

HP OpenVMS

V8.2 リリース・ノート【翻訳版】

BA322-90010

2005 年 4 月

本書は、本ソフトウェアに追加された変更点について説明します。また、インストール、アップグレード、互換性の情報についても説明します。さらに、新規および既存のソフトウェアの問題点と制限事項、ソフトウェアとドキュメントの訂正についても説明します。

改訂 / 更新情報:

新規マニュアルです。

ソフトウェア・バージョン:

OpenVMS I64 Version 8.2
OpenVMS Alpha Version 8.2

© Copyright 2005 Hewlett-Packard Development Company, L.P.

本書の著作権は日本ヒューレット・パッカート株式会社およびその子会社が保有しており、本書中の解説および図、表は日本ヒューレット・パッカート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。

また、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す。

日本ヒューレット・パッカートは、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。

Intel および Itanium は、米国およびその他の国における、Intel Corporation またはその関連会社の登録商標です。

Java は、米国における Sun Microsystems, Inc. の商標です。

Oracle は、米国における Oracle Corporation, Redwood City, California の登録商標です。

OSF および Motif は、米国およびその他の国における The Open Group の商標です。

UNIX は、The Open Group の登録商標です。

Microsoft, Windows, Windows NT, および MS Windows は、米国における Microsoft Corporation の登録商標です。

X/Open は、英国およびその他の国における X/Open Company Ltd. の登録商標です。X device は、英国およびその他の国における X/Open Company Ltd. の商標です。

原典：HP OpenVMS Version 8.2 Release Notes
© 2005 Hewlett-Packard Development Company, L.P.

本書は、日本語 VAX DOCUMENT V 2.1を用いて作成しています。

目次

まえがき	xv
1 OpenVMS ソフトウェアのインストールおよびアップグレードに関する注意事項	
1.1 弊社ソフトウェアのテクニカル・サポート方針	1-1
1.2 アプリケーションの互換性	1-3
1.3 修正キットの入手方法	1-3
1.4 ネットワーク・オプション	1-3
1.5 古いバージョンの OpenVMS と互換性のないディスク	1-4
1.6 HP DECwindows Motif for OpenVMS	1-5
1.7 HP SSL: HP SSL V1.2 のインストール	1-5
1.8 OpenVMS I64 ユーザ向けの注意事項	1-6
1.8.1 HP Integrity サーバの構成	1-6
1.8.2 Integrity Server での System Event Log (SEL) のクリア	1-7
1.8.3 Integrity サーバ用のファームウェア	1-7
1.8.4 Fibre Channel ストレージ・デバイスからのブート	1-9
1.8.5 インストール DVD からのブート	1-10
1.8.6 I64 システムのリポート設定	1-10
1.8.7 HP DECwindows Motif に関する注意事項	1-10
1.8.7.1 キーボードのサポート	1-10
1.8.7.2 サーバ起動前の周辺デバイスの接続	1-11
1.8.7.3 スタートアップ中に表示されるカウントダウン・メッセージ	1-11
1.9 OpenVMS Alpha 向けの注意事項	1-11
1.9.1 OpenVMS Alpha Version 8.2 のファームウェア	1-11
1.9.2 アップグレード・パス	1-12
1.9.3 Version 7.3-1 からのアップグレードでのエラー	1-13
1.9.4 Encryption for OpenVMS Alpha: 古いキットを削除する	1-14
1.9.5 HP DECram V3.n: アップグレード前に削除する	1-14
1.9.6 アップグレード前の Kerberos V1.0 の削除	1-15
1.9.7 アップグレード後に LANCP デバイス・データベースを変換する	1-16
1.9.8 DECnet-Plus の新しいバージョンが必要	1-16
1.9.9 SHADOW_MAX_UNIT のデフォルト設定とメモリ使用量	1-17

2	OpenVMS の関連製品に関する注意事項	
2.1	関連製品のサポート	2-1
2.2	BASIC: STARLET ライブラリの作成に V1.5A が必要 (Alpha のみ)	2-2
2.3	追加された CMAP ファイル	2-2
2.4	COBOL: I/O 実行時診断と RMS 特殊レジスタの変更	2-2
2.5	COM for HP OpenVMS (Alpha のみ)	2-3
2.5.1	COM for OpenVMS のサポート	2-3
2.5.2	高負荷時に発生するアプリケーションのレジストリ・アクセス・エラー	2-3
2.6	OpenVMS Version 8.2 では DECdfs Version 2.4 が必要	2-3
2.7	DECforms Web Connector Version 3.0 (Alpha のみ)	2-3
2.8	DEC PL/I: OpenVMS での RTL のサポート	2-4
2.9	FMS キット	2-4
2.10	Graphical Configuration Manager (GCM) (Alpha のみ)	2-5
2.11	HP DCE RPC for OpenVMS	2-5
2.11.1	DCE RPC による FAILSafe IP のサポート	2-5
2.11.2	RPC ソケットにおけるバッファ・サイズ・チューニングのサポート	2-5
2.11.3	RTI (Remote Task Invocation) RPC (I64 のみ)	2-6
2.11.4	Microsoft LAN Manager RPC はテストされていない (I64 のみ)	2-6
2.11.5	utc_multitime 係数指数の型	2-6
2.11.6	G_FLOAT 浮動小数点型と IEEE 浮動小数点型のサポート	2-6
2.12	HP DECnet-Plus for OpenVMS: X.25 データ・リンクがサポートされていない (I64 のみ)	2-8
2.13	HP DECram	2-8
2.13.1	DECram は OpenVMS Version 8.2 に含まれる	2-8
2.13.2	DECRYPT DCL コマンドとの競合	2-9
2.13.3	DECram の最大ディスク・サイズ	2-9
2.13.4	DECram コマンドとエラー	2-9
2.13.5	DECram とボリューム・シャドウイング	2-10
2.14	HP DECwindows Motif for OpenVMS	2-10
2.14.1	DECwindows のポーズ画面でロック解除できない問題 (Alpha のみ)	2-10
2.14.2	新しいロケールの追加	2-11
2.14.3	利用可能な言語バリエーション	2-11
2.14.4	LAT トランスポート・インタフェースのサポート	2-11
2.14.5	ユーザが作成したトランスポートはサポートされない	2-12
2.15	HP Secure Web Server のサポート	2-12
2.16	Pascal	2-13
2.16.1	STARLET ライブラリの作成には V5.8A (以降) が必要 (Alpha のみ)	2-13
2.16.2	アップグレード後の HP Pascal のインストール (Alpha のみ)	2-13
2.17	I64 システムでの WEBES および SEA のサポート	2-14

3	一般ユーザ向けの注意事項	
3.1	OpenVMS Freeware CD-ROM	3-1
3.1.1	フリーウェア・メニューが利用できない (I64 のみ)	3-1
3.2	オンライン・ヘルプの変更	3-2
3.3	DCL コマンド	3-3
3.3.1	ANALYZE/ERROR_LOG コマンドは OpenVMS I64 に移植されない	3-3
3.3.2	CREATE/MAILBOX: 一時的な制約	3-3
3.3.3	DIAGNOSE コマンドはサポートされない	3-3
3.3.4	SHOW LICENSE /OE および/HIERARCHY には SYSLCK 特権が必要 (I64 のみ)	3-3
3.4	Open Source Tools CD-ROM での DECmigrate の提供について	3-4
3.5	iconv の修正	3-4
3.6	HP Secure Web Browser	3-4
3.6.1	必要メモリ量の増加	3-4
3.6.2	ODS-2 ディスク・ボリュームで発生するインストール・エラー (I64 のみ)	3-5
3.7	TECO エディタは I64 システムではまだ利用できない	3-5
4	システム管理に関する注意事項	
4.1	システムのハングアップまたはクラッシュからの回復 (I64 のみ)	4-1
4.2	AUTHORIZE: DEFAULT と SYSTEM アカウントの新規クォータ	4-1
4.3	AUTOGEN: NEWPARAMS.DAT ファイルに関する新しい動作	4-2
4.4	バックアップ・ユーティリティ: 動作変更	4-3
4.5	Oracle® 8i および 9i での DECdtm/XA (Alpha のみ)	4-3
4.6	デバイス・ユニットの最大数の増加	4-3
4.7	ECP Data Collector と Performance Analyzer V5.5 (Alpha のみ)	4-4
4.8	EDIT/FDL: 推奨バケット・サイズの変更	4-4
4.9	EFI\$CP ユーティリティ: 使用は推奨できない	4-4
4.10	EFI ツール: VMS_SHOW DUMP_DEV エラー (I64 のみ)	4-5
4.11	Error Log Viewer (ELV) ユーティリティ: TRANSLATE/PAGE コマンド	4-5
4.12	外部認証	4-5
4.12.1	I64 外部認証サポート	4-5
4.12.2	DECterm 端末セッションでの SET PASSWORD の動作	4-6
4.12.3	ワークステーションではパスワードの有効期限切れは通知されない	4-6
4.13	INITIALIZE コマンドの変更	4-7
4.14	INSTALL ユーティリティ	4-7
4.15	ロック・マネージャ	4-7
4.15.1	ロック値ブロックの拡張	4-8
4.15.2	高速ロック再マスタリングと PE1	4-8
4.16	Logical Disk (LD) ユーティリティ: 問題の修正	4-9
4.17	On-Disk Structure (ODS) のレイアウトの変更	4-9

4.18	OpenVMS Cluster システム	4-10
4.18.1	OpenVMS I64 でのクラスタのサポート	4-10
4.18.2	現時点での制限事項	4-11
4.18.3	制限事項	4-11
4.18.4	クラスタの互換性のために必要なパッチ・キット	4-12
4.18.5	Fibre Channel および SCSI マルチパスと、他社製品との非互換性を修正する新しい API	4-13
4.18.6	CLUSTER_CONFIG.COM と、ルート・ディレクトリ名の制限	4-14
4.18.7	CI と LAN との間の回線切り替えによるクラスタの性能の低下	4-15
4.18.8	OpenVMS Cluster システムでの Gigabit Ethernet スイッチの制限事項	4-16
4.18.9	マルチパス・テープ・フェールオーバの制限事項	4-17
4.18.10	SCSI マルチパス媒体チェンジャでは自動フェールオーバは行われ ない	4-17
4.19	OpenVMS Galaxy (Alpha のみ)	4-18
4.19.1	Galaxy の定義	4-18
4.19.2	OpenVMS Graphical Configuration Manager	4-18
4.19.3	ES40 上の Galaxy: 非圧縮ダンプの制限事項	4-19
4.19.4	ES40 上の Galaxy: Fast Path の無効化	4-19
4.20	OpenVMS Management Station	4-19
4.21	OpenVMS Registry は Version 2 フォーマットのデータベースを壊すことがあ る	4-19
4.22	セキュリティ: DIRECTORY コマンド出力の変更	4-20
4.23	サーバ管理プロセス (SMHANDLER)	4-20
4.24	SYSGEN: セキュリティ監査の修正	4-21
4.25	SYSMAN ユーティリティ: DUMP_PRIORITY コマンドの変更	4-21
4.26	システム・パラメータ	4-22
4.26.1	新しいシステム・パラメータ	4-22
4.26.2	廃止されたシステム・パラメータ	4-22
4.26.3	廃止されたシステム・パラメータの表示	4-23
4.26.4	システム・パラメータの変更	4-23
4.26.5	MMG_CTLFLAGS: ドキュメントの誤り	4-26
4.27	Terminal Fallback Facility (TFF)	4-26
4.28	タイム・ゾーンの変更	4-28
4.29	User Environment Test Package (UETP) (I64 のみ)	4-28
4.30	推奨するキャッシュ方式	4-29
4.31	Volume Shadowing for OpenVMS	4-29
4.31.1	デバイス名の必要条件	4-29
4.31.2	DCL コマンド・プロシージャ内での SET SHADOW と SHOW SHADOW の使用についての警告	4-30
4.31.3	書き込みビットマップと異種デバイス・シャドウイング (DDS) の注意事 項	4-30
4.31.4	KZPDC (Smart Array 5300) の制限事項	4-32
4.31.5	シャドウ・セット・マージ遅延の算出方法の変更	4-33
4.31.6	ANALYZE/DISK/SHADOW コマンドの動作	4-33
4.31.7	異種デバイス・シャドウ・セットでの ANALYZE/DISK/SHADOW コマ ンドの動作	4-34

4.31.8	/MINICOPY を使用したシャドウ・セット・メンバのディスマウント	4-35
5	プログラミングに関する注意事項	
5.1	プログラムの再コンパイルとリンクが必要 (I64 のみ)	5-1
5.2	特権プログラムの再コンパイルが必要 (Alpha のみ)	5-1
5.3	特権データ構造体の変更	5-2
5.3.1	KPB 拡張	5-3
5.3.2	CPU の名前空間	5-3
5.3.3	64 ビットの論理ブロック番号 (LBN)	5-3
5.3.4	動的スピンロックのフォーク	5-3
5.3.5	UCB と DDB のアップデート	5-4
5.3.6	PCB\$T_TERMINAL のサイズの拡張	5-5
5.3.7	スレッド単位のセキュリティは特権付きコードとデバイス・ドライバに影響する	5-5
5.3.8	OpenVMS フォーク・スレッド作成のための IPL 要件の強制	5-6
5.4	浮動小数点型データを使用するアプリケーション	5-7
5.5	Ada コンパイラはまだ利用できない (I64 のみ)	5-8
5.6	Backup API: ジャーナリング・コールバック・イベントの制限事項	5-8
5.7	C プログラム: CASE_LOOKUP=SENSITIVE を設定したコンパイル	5-8
5.8	C ランタイム・ライブラリ	5-9
5.8.1	socket_fd を使用するプログラムでのメモリ・リークの修正	5-9
5.8.2	vsnprintf と snprintf によるユーザ・バッファの上書きの修正	5-9
5.8.3	mmap と mprotect の変更	5-9
5.8.4	getpwnam_r と getpwuid_r のポインタの問題の修正	5-10
5.8.5	_strtok_r32 と _strtok_r64 がスコープ内になった	5-10
5.8.6	iconv プロトタイプの , const 型修飾子の追加 (Alpha のみ)	5-10
5.8.7	<stddef.h>ヘッダの修正 (Alpha のみ)	5-11
5.8.8	getc マクロの引数が括弧で保護されるようになった (Alpha のみ)	5-11
5.8.9	インライン関数 getc および getchar の CXXL 接頭辞の問題の修正 (Alpha のみ)	5-11
5.8.10	std 名前空間での非 std 関数の宣言 (Alpha のみ)	5-11
5.8.11	大きなファイル・オフセットでの lseek の問題の修正 (Alpha のみ)	5-12
5.8.12	<errno.h>内の新しい EABANDONED コード	5-12
5.8.13	mktime の問題の修正	5-12
5.8.14	<poll.h>内の POLLWRNORM が POLLOUT と等価になった	5-13
5.8.15	<in6.h>内の IPV6 構造体がパック形式になった	5-13
5.8.16	<builtins.h>内の __PAL_BUGCHK の問題の修正	5-13
5.8.17	C++ コンパイラの statvfs に関するエラーの修正	5-13
5.8.18	glob と globfree の問題の修正	5-14
5.8.19	DECC\$SHR_EV56 が正しくリンクされた	5-14
5.8.20	Zone Information Compiler (zic) のアップデート	5-14
5.9	呼び出し標準規則とローテートするレジスタ (I64 のみ)	5-15
5.10	Common Data Security Architecture (CDSA) に関する考慮	5-15
5.10.1	Secure Delivery Advanced Developer's Kit	5-15
5.10.2	インストールと初期化に関する考慮	5-16
5.11	照合キーを指定する CONVERT/NOSORT で発生する可能のある CONVERT-I-SEQ エラー	5-17

5.12	デバッグ・モード: CPUSPINWAIT バグ・チェックの回避	5-17
5.13	Delta/XDelta デバッグ	5-18
5.13.1	Delta デバッグは OpenVMS I64 では利用できない	5-18
5.13.2	I64 システムでの XDelta の制限事項	5-18
5.13.3	I64 システムと Alpha システムでの XDelta の相違点	5-19
5.13.4	XDelta のレジスタ表示に関する考慮 (I64 のみ)	5-19
5.14	ファイル・アプリケーション: 『Guide to OpenVMS File Applications』の訂正	5-19
5.15	RMS 構造体についての HP BLISS コンパイラの警告 (I64 のみ)	5-20
5.16	HP COBOL ランタイム・ライブラリ (RTL)	5-21
5.16.1	SET および COB\$SWITCHES	5-21
5.16.2	レコード・ロックの問題の修正	5-21
5.17	HP Decimal Support ランタイム・ライブラリ (RTL)	5-21
5.18	I64 用の HP Fortran	5-22
5.19	OpenVMS 用 HP MACRO	5-23
5.19.1	OpenVMS I64 用 HP MACRO	5-23
5.19.2	/TIE 修飾子のデフォルトは Alpha と I64 で異なる	5-23
5.19.3	/OPTIMIZE=VAXREGS 修飾子は I64 ではサポートされない	5-23
5.19.4	CODGENWARN メッセージは無視できる (Alpha のみ)	5-24
5.19.5	操作単位のサポート (I64 のみ)	5-24
5.19.6	デフォルトでは整数のゼロ除算エラーは検出されない (I64 のみ)	5-24
5.19.7	整数を負の最大値で除算するとコンパイラがクラッシュする (I64 のみ)	5-24
5.19.8	整数の除算で“V”コンディション・コードが正しくセットされない (I64 のみ)	5-25
5.19.9	浮動小数点数のゼロ除算エラーが検出されない (I64 のみ)	5-25
5.19.10	INSV 命令でメモリを余計に上書きする (I64 のみ)	5-25
5.20	Hypersort ユーティリティ	5-26
5.20.1	弊社への問題の報告	5-26
5.20.2	ラージ・ファイルの制限事項	5-26
5.20.3	Hypersort と VFC ファイルの制限事項	5-27
5.20.4	/FORMAT=RECORD_SIZE の制限事項	5-27
5.20.5	Hypersort と検索リスト, および論理名の使用	5-27
5.20.6	作業ファイルの空き領域不足	5-27
5.20.7	入力アスタリスク(*)の制限事項	5-28
5.20.8	最適化されたワーキング・セット・エクステントとページ・ファイル・クォータの設定	5-28
5.21	Intel®アセンブラ (I64 のみ)	5-28
5.22	Librarian ユーティリティ	5-28
5.22.1	data-reduced ELF オブジェクト・ライブラリとのリンクは推奨できない (I64 のみ)	5-28
5.22.2	I64 ライブラリへの.STB ファイルの挿入または置き換えの失敗 (I64 のみ)	5-29
5.22.3	プロセス・クォータが低すぎると Librarian がエラーを通知しない問題	5-29
5.22.4	オブジェクト・モジュール名の長さの問題 (I64 のみ)	5-30
5.23	Lightweight Directory Access Protocol (LDAP) API	5-30
5.23.1	ld が NULL の場合の ldap_get_option ルーチンからのエラー戻り値 (-1)	5-30

5.23.2	ber_flatten()ルーチンが中括弧の不一致を検出しない	5-30
5.24	OpenVMS Alpha 用 Linker ユーティリティ	5-30
5.24.1	LINK/NATIVE_ONLY のヘルプ・テキストの明確化	5-31
5.24.2	多数のファイルを指定した場合に Linker がハングアップしたように見える	5-31
5.24.3	ライブラリ・チェックにおける Linker のデフォルト動作の変更	5-33
5.24.4	スタックのエレメント数は最大 25 に制限	5-33
5.25	OpenVMS I64 用 Linker ユーティリティ	5-33
5.25.1	I64 リンカと Alpha リンカの違い	5-34
5.25.2	LINK_ORDER セクション・ヘッダ・フラグはサポートされていない	5-34
5.25.3	data-reduced ELF オブジェクト・ライブラリとのリンクは推奨できない	5-34
5.25.4	マップの Image Synopsis セクションの誤り	5-35
5.25.5	DIFTYPE および RELODIFTYPE メッセージ	5-35
5.26	LTDRIVER: CANCEL SELECTIVE の制限事項	5-37
5.27	Mail ユーティリティ: 呼び出し可能メールのスレッドの制限事項	5-37
5.28	OpenVMS Debugger	5-38
5.28.1	全般的な問題点と回避方法 (I64 のみ)	5-38
5.28.2	Basic 言語での問題 (I64 のみ)	5-39
5.28.3	C++ 言語での問題 (I64 のみ)	5-39
5.28.4	COBOL 言語での問題 (I64 のみ)	5-39
5.28.5	Fortran 言語での問題 (I64 のみ)	5-39
5.28.6	Pascal 言語での問題 (I64 のみ)	5-40
5.28.7	SET SCOPE コマンド: 動作の変更	5-40
5.28.8	SHOW IMAGE コマンドの変更	5-40
5.28.9	クライアント/サーバ・インタフェース: 以前のバージョンはサポートされない (Alpha のみ)	5-40
5.29	OpenVMS のシステム・ダンプ・アナライザ (SDA)	5-41
5.29.1	READ コマンドのデフォルトの変更	5-41
5.29.2	CLUE コマンドは OpenVMS I64 に移植されていない	5-41
5.29.3	SHOW CALL_FRAME 機能の変更 (I64 のみ)	5-41
5.30	OpenVMS I64 Version 8.2 に含まれない PL/I ライブラリ	5-42
5.31	POSIX スレッド・ライブラリ	5-43
5.31.1	例外処理中のスタック・オーバフロー (I64 のみ)	5-43
5.31.2	I64 システムでの THREADCP コマンドの動作	5-44
5.31.3	浮動小数点のコンパイルと例外 (I64 のみ)	5-44
5.31.4	C 言語コンパイル・ヘッダ・ファイルの変更	5-44
5.31.5	新しい優先順位調整アルゴリズム	5-45
5.31.6	プロセス・ダンプ	5-46
5.31.7	動的 CPU 構成の変更	5-46
5.31.8	デバグガ計測機能は動作しない	5-47
5.32	RMS: グローバル・バッファ VA 管理の改善	5-47
5.33	RMS Journaling	5-47
5.33.1	カーネル・スレッドと互換性のないリカバリ・ユニット・ジャーナリング	5-47
5.33.2	ジャーナル・ファイル作成の変更	5-48
5.33.3	OSI 環境でのリカバリ・ユニット・ジャーナリングされたファイルへのリモート・アクセス	5-48

5.33.4	順方向 (AI) ジャーナリング	5-49
5.33.5	VFC 形式の順編成ファイル	5-49
5.34	RTL ライブラリ (LIB\$)	5-50
5.34.1	RTL ライブラリ (LIB\$) のヘルプ	5-50
5.34.2	RTL Library (LIB\$): 呼び出し標準ルーチン (I64 のみ)	5-50
5.35	Screen Management (SMG\$) のドキュメント	5-51
5.36	SORT32 ユーティリティ	5-52
5.36.1	DFS サービス・ディスクでの CONVERT の問題	5-52
5.36.2	一時作業ファイルが削除されないことがある	5-52
5.36.3	複合条件のある SORT/SPECIFICATION: 要件	5-52
5.36.4	可変長レコードでの性能の問題	5-53
5.36.5	作業ファイル・ディレクトリの制約事項	5-53
5.37	System Code Debugger (SCD) はまだ I64 システムで使用できない	5-53
5.38	システム・サービス	5-53
5.38.1	\$GETJPI の項目コード SCHED_CLASS_NAME の記述誤り	5-53
5.38.2	PFN マップ・セクションに必要となる新しい GETSYI 項目コード (I64 のみ)	5-54
5.38.3	PFN マップ・セクションとキャッシュされないメモリ (I64 のみ)	5-54
5.38.4	SYS\$ACM: I64 システムでの使用	5-55
5.38.5	I64 では SYS\$GOTO_UNWIND は利用できない	5-55
5.39	タイマ・キュー・エントリ (TQE)	5-56
5.40	Traceback 機能	5-56
5.40.1	API エラー (I64 のみ)	5-56
5.40.2	問題の修正	5-57
5.41	Watchpoint ユーティリティ (I64 のみ)	5-57
5.42	プログラム全体の浮動小数点モード (I64 のみ)	5-57
6	ハードウェアに関する注意事項	
6.1	MP コンソールと BMC コンソールの制約事項 (I64 のみ)	6-1
6.1.1	入力デバイス, 出力デバイス, およびエラー・デバイスの制約事項	6-2
6.1.2	削除キーへの Ctrl/H の再マッピング	6-2
6.2	AlphaServer 2100	6-2
6.2.1	コンソール表示	6-3
6.2.2	SCSI コントローラの制限事項	6-4
6.3	AlphaServer 8200/8400: FRU テーブル・エラー	6-4
6.4	AlphaServer ES47/ES80/GS1280 システム	6-4
6.4.1	ES47/ES80/GS1280 ソフト・パーティションでの INIT コンソール・コマンドの使用	6-4
6.4.2	SYSGEN パラメータ PHYSICAL_MEMORY の設定	6-5
6.4.3	RAD のサポート	6-5
6.4.4	ライセンス要件	6-5
6.4.5	STOP/CPU およびシャットダウン動作	6-5
6.4.6	MBM での時刻の設定	6-6
6.5	AlphaServer GS シリーズ・システム	6-6
6.5.1	AlphaServer GS80/160/320 システム: デバイスの制限事項	6-6
6.5.2	OpenVMS Galaxy ライセンスの実行	6-6
6.5.3	ライセンスのインストール	6-7

6.5.4	AlphaServer GS60/GS60E/GS140 複数 I/O ポート・モジュール構成の制限事項	6-9
6.6	AlphaStation 200/400: ISA_CONFIG.DAT の変更が必要	6-9
6.7	AlphaStation 255: PCI 構成の制限事項	6-10
6.8	ATI RADEON 7000 グラフィック (I64 のみ)	6-10
6.8.1	I64 のグラフィック・サポート	6-11
6.8.2	RADEON 7000 ではハードウェア・アクセラレーション 3D グラフィックはサポートされていない	6-11
6.9	ATI RADEON 7500 グラフィック	6-11
6.9.1	リソースの要件	6-11
6.9.2	Version 8.2 の新機能	6-12
6.9.2.1	テキストその他のビットマップの描画性能の改善	6-12
6.9.2.2	DMA 操作の性能改善	6-13
6.9.2.3	OpenGL テクスチャ・コピーの性能改善	6-13
6.9.2.4	画面解像度 1024 × 864 の追加	6-13
6.9.2.5	ハードウェア・アクセラレーションによるインダイレクト 3D レンダリング	6-13
6.9.3	問題の修正 (Alpha のみ)	6-13
6.9.4	DECW\$OPENGLSHR_RADEON.EXE を DECW\$MESA3DSHR_RADEON.EXE にリネーム	6-14
6.9.5	サポートの制限事項	6-15
6.9.6	高リフレッシュ・レートでの画像への影響	6-15
6.9.7	OpenGL は IEEE 演算のみをサポート	6-15
6.9.8	グラフィック・コンソール (OPA0) に出力する際に DECwindows サーバがハングアップ	6-15
6.9.9	モニタは初期化時に接続されていなければならない	6-16
6.9.10	ブート・リセットの推奨 (Alpha のみ)	6-16
6.9.11	オーバーレイ・プレーンはサポートされない	6-16
6.9.12	単一カラーマップのみサポート	6-16
6.9.13	すべてのウィンドウで同じビット深度	6-17
6.9.14	読み取り/書き込みカラー・マップのピクセル深度	6-17
6.9.15	backing store と save unders は 3D ウィンドウをサポートしていない	6-17
6.9.16	スレッドの制限事項	6-18
6.9.17	シングル・バッファ・ビジュアルがサポートされていない	6-18
6.9.18	カラー・インデックス・モードでの 3D がサポートされていない	6-18
6.9.19	タイマ・メカニズム	6-18
6.10	DECwindows X11 ディスプレイ・サーバ (Alpha のみ)	6-19
6.10.1	社名とオペレーティング・システム・バージョン情報の更新	6-19
6.10.2	S3 マルチヘッド・グラフィック	6-19
6.10.3	グラフィック・ボードのサポート	6-20
6.11	DIGITAL Modular Computing Components (DMCC)	6-20
6.11.1	Alpha 5/366 および 5/433 PICMG SBC の制限事項	6-20
6.11.2	SRM コンソールの更新	6-20
6.12	Digital Personal Workstation: OpenVMS V7.3-1 およびそれ以降のブート	6-21
6.13	I/O 負荷が重い場合, LUN が多数あるデュアル・コントローラ HSGnn に障害が発生することがある	6-21
6.14	Open3D グラフィックのライセンス方式の変更	6-22
6.15	OpenVMS 用の PowerStorm 300/350 PCI グラフィック・サポート	6-23

6.16	RFnn DSSI ディスク・デバイスとコントローラ・メモリのエラー	6-24
6.17	RZnn ディスク・ドライブの考慮事項	6-26
6.17.1	RZ25M と RZ26N ディスク・ドライブ: 推奨事項	6-26
6.17.2	RZ26N および RZ28M ディスク: 推奨ファームウェア・サポート	6-27
6.17.3	RZ26L および RZ28 ディスク: マルチホストで使用するために必要なファームウェア	6-27
6.17.3.1	ファームウェア・リビジョン・レベル 442 の必要条件	6-27
6.17.3.2	ファームウェア・リビジョン・レベル 442 のインストール手順	6-28
6.18	ZLX グラフィック・ボードのサポート	6-28
6.19	OpenVMS デバイス・ドライバの再コンパイルと再リンク	6-29
6.19.1	Alpha および VAX の SCSI デバイス・ドライバ	6-29
6.19.2	OpenVMS Alpha デバイス・ドライバ	6-29
6.20	MON バージョンのデバイス・ドライバの処理	6-30
6.21	スレッド単位のセキュリティが Alpha デバイス・ドライバに与える影響	6-30
6.22	OpenVMS Alpha ドライバのデバイス IPL の設定	6-30
6.23	CRCTX ルーチンの機能の強化	6-31
A	リタイア製品情報	
A.1	Compaq Open3D レイヤード・プロダクトは Version 8.2 ではサポートされない	A-1
A.2	Open3D グラフィックのライセンス方式の変更	A-2
A.3	DECamds は OpenVMS Version 8.2 ではサポートされない	A-2
A.4	DECevent はサポートされない	A-2
A.5	2004 年 12 月に DECwrite のサービス期間が終了	A-3
A.6	Error Log Report Formatter (ERF) のサポート終了	A-3
A.7	ISA_CONFIG.DAT のサポートの将来のリリースでの中止	A-4
A.8	POSIX 1003.4a Draft 4 インタフェースのサポート中止	A-4
A.9	ビジュアル・スレッド: Version 8.2 ではサポートされない	A-4
A.10	アーカイブ扱いのマニュアル	A-4
B	インターロックされたメモリ命令の使用 (Alpha のみ)	
B.1	必要なコード・チェック	B-1
B.2	コード分析ツール (SRM_CHECK) の使用	B-2
B.3	規則に準拠していないコードの特徴	B-3
B.4	コーディングの必要条件	B-4
B.5	コンパイラのバージョン	B-6
B.6	ALONONPAGED_INLINE または LAL_REMOVE_FIRST によるコードの再コンパイル	B-7

索引

表

1-1	サポートされている DECwindows Motif のバージョン	1-5
4-1	クラスタの互換性のために必要なパッチ・キット	4-13
4-2	Galaxy の定義	4-18
4-3	TFF 文字フォールバック・テーブル	4-27
6-1	デバイス記述ブロックの変更点	6-10
6-2	サポートされるマイクロコードのレベル	6-24
6-3	DSSI ディスク・デバイスのマイクロコードを更新するコマンド	6-26
6-4	リビジョン・レベル 442 ファームウェアの互換性	6-28
B-1	OpenVMS コンパイラのバージョン	B-6

対象読者

本書は、HP OpenVMS Alpha または HP OpenVMS I64 Version 8.2 オペレーティング・システムのすべてのユーザを対象にしています。OpenVMS Version 8.2 のインストール、アップグレード、あるいはご使用前にお読みください。

本書の構成

本書の構成は以下のとおりです。

- 第 1 章は、OpenVMS Alpha 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と、OpenVMS I64 のインストールに関する注意事項について説明します。
- 第 2 章は、OpenVMS の関連製品のインストールおよびサポート情報について説明します。
- 第 3 章は、OpenVMS オペレーティング・システムの一般的な使用に関する注意事項について説明します。
- 第 4 章は、特に OpenVMS のシステム管理に関する注意事項について説明します。
- 第 5 章は、OpenVMS システムでのプログラミングに関する注意事項について説明します。コンパイラ、リンカ、およびランタイム・ライブラリ・ルーチンを使用する際の注意事項も含まれています。
- 第 6 章は、OpenVMS オペレーティング・システムが実行されるハードウェアについての情報と、OpenVMS のデバイス・サポートについての情報を示します。
- 付録 A は、本リリースではサポートされなくなった OpenVMS 製品、または廃止予定の OpenVMS 製品についての情報を示します。
- 付録 B は、プロセッサ Alpha 21264 (EV6) を使用する際に非常に重要な、インターロックされたメモリ命令の正しい使用方法について説明します。

可能な限り、機能または製品名ごとに注意事項が分類されています。

本書に記載されている注意事項には、今回のリリースから適用されるものと、以前のリリースから引き続き適用されるものが含まれています。各注意事項が最初に適用されたバージョンが、項目タイトルとして (V7.3-2 のように) 示されています。または、古い注意事項が更新された場合はそのバージョンが示されています (たとえば、

V7.3-2 の注意事項で、Version 8.2 で改訂されたものには、V8.2 という見出しが付きます。

次のような場合も、以前のバージョンの注意事項が掲載されます。

- その情報が OpenVMS ドキュメント・セットのどのマニュアルにも記載されておらず、その注意事項の内容がまだ適用される場合。
- 注意事項がマルチ・バージョンの OpenVMS Cluster システムに適用される可能性がある場合。

関連資料

本バージョンの OpenVMS オペレーティング・システムで利用できるドキュメントの一覧については、『HP OpenVMS Version 8.2 新機能説明書』を参照してください。

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、キー名が四角で囲まれている場合には、キーボード上でそのキーを押すことを示します。テキストの中では、キー名は四角で囲まれていません。 HTML 形式のドキュメントでは、キー名は四角ではなく、括弧で囲まれています。
...	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。• 前出の 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任意のオプションです。オプション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。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、大括弧に囲まれた要素は省略できません。

表記法	意味
[]	コマンド形式の説明では、括弧内の要素を分けている垂直棒線はオプションを 1 つまたは複数選択するか、または何も選択しないことを意味します。
{ }	コマンドの形式の説明において、中括弧で囲まれた要素は必須オプションです。いずれか 1 つのオプションを指定しなければなりません。
太字	太字のテキストは、新しい用語、引数、属性、条件を示しています。また、変数を示す場合にも使用されています。
<i>italic text</i>	イタリック体のテキストは、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 <i>number</i>)、コマンド行 (たとえば <i>/PRODUCER=name</i>)、コマンド・パラメータ (たとえば <i>device-name</i>) などの変数を示す場合にも使用されます。
UPPERCASE TEXT	英大文字のテキストは、コマンド、ルーチン名、ファイル名、ファイル保護コード名、システム特権の短縮形を示します。
Monospace type	モノスペース・タイプの文字は、コード例および会話型の画面表示を示します。 C プログラミング言語では、テキスト中のモノスペース・タイプの文字は、キーワード、別々にコンパイルされた外部関数およびファイルの名前、構文の要約、または例に示される変数または識別子への参照などを示します。
-	コマンド形式の記述の最後、コマンド行、コード・ラインにおいて、ハイフンは、要求に対する引数が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。

OpenVMS ソフトウェアのインストールおよびアップグレードに関する注意事項

この章では、OpenVMS I64 Version 8.2 のインストール、および OpenVMS Alpha Version 8.2 のインストールやアップグレードに必要な情報について説明します。Alpha および I64 システムのユーザに共通のトピックを最初に説明し、その後の節で、特定のプラットフォームのユーザ向けに説明します。

OpenVMS Version 8.2 のインストールまたはアップグレードの際には次のマニュアルをすべてお読みください。

- 『HP OpenVMS Version 8.2 リリース・ノート[翻訳版]』（本書）
- 『HP OpenVMS Version 8.2 新機能説明書』
- 『HP OpenVMS Version 8.2 Upgrade and Installation Manual』

ハードウェアに関する注意事項については第 6 章、関連製品については第 2 章を参照してください。

1.1 弊社ソフトウェアのテクニカル・サポート方針

弊社では、OpenVMS オペレーティング・システム・ソフトウェアの最新版（現在出荷されているバージョン）と、直前のバージョンの製品について、ソフトウェアのテクニカル・サポートを行います。各バージョンは、リリース日より 24 カ月間、または次の次のバージョンがリリースされるまでの期間（いずれか長い方）、サポートされます。「バージョン」は、新しい機能と拡張を含むリリースと定義されます。サポート方針としては、パッチ・キットやメンテナンス専用のリリースは、この「バージョン」の定義に当てはまりません。

これらのガイドラインに基づいて、OpenVMS のバージョンごとに、OpenVMS オペレーティング・システム・ソフトウェアの、現行バージョン・レベルのサポート（標準サポート (SS)）と旧バージョンのサポート (PVS) が行われます。OpenVMS Alpha と OpenVMS VAX の最近のバージョンに対する現在のサポート・レベルは、次の Web ページから入手してください。

http://h71000.www7.hp.com/openvms/openvms_supportchart.html

OpenVMS ソフトウェアのインストールおよびアップグレードに関する注意事項

1.1 弊社ソフトウェアのテクニカル・サポート方針

オペレーティング・システムのサポート方針は、すべての OpenVMS メジャー・リリース、新機能リリース、および拡張リリースに適用されます。各リリースの定義は、次のとおりです。

- メジャー・リリースには、根本的な新機能が含まれます。バージョン番号の整数部分に、次に大きい数字が使用されます (例: 6.2-1H1 から 7.0 へ)。

アプリケーションへの影響: OpenVMS の内部インタフェースが変更されています。大半のアプリケーションではバイナリ互換性が維持されますが、独立系ソフトウェア・ベンダ (ISV) は、新しいバージョンでテストを行い、場合によっては新しいアプリケーション・キットをリリースする必要があります。アプリケーション・パートナーは、オペレーティング・システムの新しい機能を活用するために、新しいアプリケーション・キットをリリースすることもできます。

- 新機能リリースには、新機能、拡張機能、およびメンテナンス上の更新が含まれます。バージョン番号の小数第 1 位に、次に大きい数字が使用されます (例: 7.2 から 7.3 へ)。

アプリケーションへの影響: このリリースでは、公開されたアプリケーション・プログラミング・インタフェース (API) は廃止されません。ただし、重要な新機能の追加や性能の向上により、OpenVMS の内部インタフェースが変更されていることがあります。マニュアルに記載されている API を使用するアプリケーションのほとんど (95%) については、新しいアプリケーション・キットは不要です。デバイス・ドライバとカーネル・レベルのアプリケーション (標準で規定されていない API または説明されていない API を使用するアプリケーション) については、適合性テストが必要になることがあります。

- 拡張リリースには、既存の機能に対する拡張機能やメンテナンス上の更新が含まれます。バージョン番号のダッシュ (-) 以降に、次に大きい数字が使用されます (例: 7.3-2)。

アプリケーションへの影響: このリリースには、新しいハードウェア・サポート、ソフトウェアの拡張機能、およびメンテナンス上の更新が含まれていますが、この変更は公開された API を使用するアプリケーションに影響を及ぼしません。ISV では新しいリリースをテストしたり、新しいアプリケーション・キットを作成したりする必要はありません。

- ハードウェア・リリースは、特定のハードウェア・サポートを含む次のリリースが出されてから 12 カ月間、現行バージョン・レベルでサポートされます。ハードウェア・リリースは、新しいハードウェアを購入いただいた場合にのみ送付されます。サービス契約だけの場合は、ご利用いただけません。

OpenVMS の次の主な製品は、同時に出荷されたオペレーティング・システムのバージョンと同じ期間だけ同じレベル (標準サポートまたは旧バージョンのサポート) でサポートされます。

- HP Advanced Server for OpenVMS
- HP DECnet (フェーズ IV)

- HP DECnet-Plus for OpenVMS
- HP OpenVMS Cluster Client Software
- HP OpenVMS Cluster Software for OpenVMS
- HP PathWorks または HP PATHWORKS for OpenVMS
- HP RMS Journaling for OpenVMS
- HP TCP/IP Services for OpenVMS
- HP Volume Shadowing for OpenVMS

これらの製品のサポート・サービスを依頼するには、個別のサポート契約が必要です。オペレーティング・システムのサポート契約には含まれていません。

1.2 アプリケーションの互換性

OpenVMS では、公開された API は、どのリリースでも一貫してサポートされます。通常、公開された API を使用するアプリケーションであれば、OpenVMS の新しいリリースをサポートするために変更が必要になることはありません。廃止された API は、マニュアルから削除されますが、OpenVMS では引き続き API として使用できます。

1.3 修正キットの入手方法

弊社製品の修正キットは、HP IT リソース・センタ (ITRC) からオンラインで入手できます。ITRC パッチ・ダウンロード・サイトを使用するには、ユーザ登録とログインが必要です。すべてのユーザが登録可能で、サービス契約は不要です。次の URL で、登録とログインができます。

<http://www2.itrc.hp.com/service/patch/mainPage.do>

また、FTP を使用して、次の場所からパッチを入手することもできます。

ftp://ftp.itrc.hp.com/openvms_patches

1.4 ネットワーク・オプション

V8.2

OpenVMS では、使用するネットワーク・プロトコルを柔軟に選択できます。DECnet が必要な場合も、TCP/IP が必要な場合も、OpenVMS ではネットワークにとって最適なプロトコルあるいは複数のプロトコルの組み合わせを選択できます。OpenVMS では、弊社のネットワーク製品と他社製ネットワーク製品のどちらも使用できます。

OpenVMS Version 8.2 のインストール時に、サポートされている次の HP ネットワーク・ソフトウェアをインストールすることができます。

- HP DECnet-Plus Version 8.2 for OpenVMS または HP DECnet フェーズ IV Version 8.2 for OpenVMS (2 つの DECnet 製品を同時に実行することはできません)

DECnet-Plus には、DECnet フェーズ IV 製品のすべての機能に加えて、TCP/IP または OSI プロトコルを介して DECnet を実行する機能も含まれています。

Prior Version Support 契約を結ばれているお客様に対しては、DECnet フェーズ IV がサポートされます。Prior Version Support サービスについての詳細は、第 1.1 節を参照してください。

- HP TCP/IP Services for OpenVMS Version 5.5

TCP/IP Services と DECnet は、システム上で同時に実行できます。システムに HP DECnet-Plus for OpenVMS と TCP/IP Services をインストールすると、DECnet アプリケーション、OSI アプリケーション、またはその両方を、TCP/IP ネットワーク上で実行できます。TCP/IP 上での DECnet の実行 (RFC 1859)、TCP/IP 上での OSI の実行 (RFC 1006) についての詳細は、『DECnet-Plus for OpenVMS Management Guide』を参照してください。

または、OpenVMS をインストールした後で、OpenVMS Version 8.2 で動作する他社製ネットワーク製品をインストールすることもできます。

インストールの後で HP ネットワーキング・ソフトウェアを構成したり管理したりする方法については、TCP/IP、DECnet-Plus、または DECnet の各マニュアルを参照してください。

1.5 古いバージョンの OpenVMS と互換性のないディスク

V8.2

OpenVMS Version 8.2 のインストール手順では、ターゲット・ディスクをボリューム拡張付き (INITIALIZE/LIMIT) で初期化します。これにより、Version 7.2 よりも前の OpenVMS とはディスクの互換性がなくなります。多くの場合、特に問題は発生しません。ただし、この新しいディスクを Version 7.2 よりも前の OpenVMS にマウントする場合は、そのバージョンのオペレーティング・システムと互換性を持つように設定しなければなりません。設定手順の詳細は、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

これらの手順を実行すると、新しいシステム・ディスクの最小割り当てサイズ (/CLUSTER_SIZE で定義) は、以前より大きくなる場合があります。その結果、小さなファイルが必要以上のスペースを専有するようになります。そのため、この手順は、Version 7.2 より前の OpenVMS にマウントしなければならないシステム・ディスクに対してのみ実行してください。

注意

ODS-5 ディスクも， Version 7.2 より前の OpenVMS とは互換性がありません。

1.6 HP DECwindows Motif for OpenVMS

V8.2

次の表に， HP OpenVMS 8.2 オペレーティング・システムでサポートされている DECwindows Motif のバージョンを示します。

表 1-1 サポートされている DECwindows Motif のバージョン

OpenVMS のバージョン	DECwindows Motif のバージョン
OpenVMS I64 8.2	DECwindows Motif for OpenVMS I64 V1.5
OpenVMS Alpha 8.2	DECwindows Motif for OpenVMS Alpha V1.5

DECwindows Motif ソフトウェアは，特定のバージョンの OpenVMS サーバおよびデバイス・ドライバのイメージを必要とします。表 1-1 に示すように，オペレーティング・システム環境に適したバージョンの DECwindows Motif をインストールまたはアップグレードするようにしてください。

以前のバージョンの DECwindows Motif のサポート情報については，『日本語 HP DECwindows Motif for OpenVMS リリース・ノート』を参照してください。

DECwindows Motif ソフトウェアのインストールについての詳細は，『日本語 HP DECwindows Motif for OpenVMS インストレーション・ガイド』を参照してください。

1.7 HP SSL: HP SSL V1.2 のインストール

V8.2

Alpha システムまたは I64 システムに HP SSL V1.2 をインストールすると，openssl.org からの最新のセキュリティ修正のほか，新しい暗号方式やアルゴリズムが利用できます。HP SSL V1.2 キットは，レイヤード・プロダクト・ディストリビューションに含まれています。オペレーティング・システムと TCP/IP Serices for OpenVMS をインストールまたはアップグレードした後に，HP SSL V1.2 をインストールすることをお勧めします。

SSL V1.2 をインストールする前に、SSL のそれ以前のバージョンは削除しておく必要があります。SSL の削除とインストールには、以下のコマンドを使用してください。

```
$ PRODUCT REMOVE SSL  
$ PRODUCT INSTALL SSL /SOURCE=ddcu:[directory]
```

SSL のインストールと使用法の詳細については、『HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS』を参照してください。

1.8 OpenVMS I64 ユーザ向けの注意事項

以降の説明は、主に OpenVMS I64 システムのユーザ向けです。

1.8.1 HP Integrity サーバの構成

V8.2

OpenVMS I64 Version 8.2 リリースは、rx1600-2, rx1620-2, rx2600-2, rx2620-2, および rx4640-8 Integrity サーバのすべての標準システム機能とコア I/O をサポートしています。

OpenVMS I64 Version 8.2 には、2 つの主要なストレージ・インターコネクタがあります。

- SCSI ストレージ・タイプは U320 です。これは一部の Integrity Server の Core I/O で使用しており、また A7173A U320 SCSI アダプタでも使用しています。外部の SCSI ストレージとの接続については、ストレージ・シェルフとして DS2100 と MSA30 をサポートしています。
- 外部の Fibre Channel ストレージとの接続には、dual-port 2GB Fibre Channel Universal PCI-X アダプタ (A6826A) を使用します。このアダプタは、OpenVMS がサポートしている SAN ベースの Fibre Channel ストレージのいずれとも接続できます。

以前の評価版キットや、フィールド・テスト版キットをご使用のユーザは、下記の重要な考慮事項にご注意ください。

- SCSI アダプタのうち、U160 アダプタ (A6829A) は、OpenVMS I64 Version 8.2 では公式にサポートしていませんし、2005 年中にサポート期間が終わります。ただし、システムを現在の構成のままにしておく限り、このアダプタを既存のハードウェア用に使用することは可能です。アダプタを追加したり、別のサーバに移行する場合は、U320 タイプの SCSI アダプタが必要になります。

- Fibre Channel については、AB232A または KGPSA-EA FC アダプタをご使用の場合があります。これらのアダプタは、OpenVMS I64 Version 8.2 ではサポートされませんので、Version 8.2 上で製品版のアプリケーションを実行する前に、A6826A FC アダプタにアップグレードする必要があります。

1.8.2 Integrity Server での System Event Log (SEL) のクリア

V8.2

HP Integrity サーバは、システム・コンソールのストレージ内に System Event Log (SEL) を保存しており、OpenVMS I64 システムは自動的に SEL の内容を OpenVMS エラー・ログに転送します。コンソールから操作しているとき、正常なブート処理中に BMC (Baseboard Management Controller) SEL が満杯である旨のメッセージが表示されることがあります。BMC SEL が満杯の場合でも、プロンプトに従って操作すれば、OpenVMS は SEL の内容を処理し、運用を継続できます。手動で SEL をクリアしたい場合は、EFI シェルのプロンプトで、次のコマンドを入力してください。

```
Shell> clearlogs SEL
```

このコマンドにより、SEL の内容がクリアされます。このコマンドは、最新のシステム・ファームウェア・バージョンで使用できます。

ご使用の Integrity サーバに MP (Management Processor) が構成されていて、MP コンソールと接続しているときに BMC イベント・ログの警告メッセージが表示された場合は、MP を使用して BMC イベント・ログをクリアすることができます。Ctrl/B を押して、MP>プロンプトを表示させます。MP>プロンプトで(メイン・メニューから) SL を入力し、C オプションでログをクリアします。

最新のシステム・ファームウェアをロードして使用することをお勧めします。システム・ファームウェアのアップデートの詳細については『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

1.8.3 Integrity サーバ用のファームウェア

V8.2

OpenVMS I64 Version 8.2 は、サポートされている各 Integrity サーバ用の最新リリースのファームウェアでテストされています。次の表に、推奨するファームウェア・バージョンの一覧を示します。

OpenVMS ソフトウェアのインストールおよびアップグレードに関する注意事項

1.8 OpenVMS i64 ユーザ向けの注意事項

システム	システム・ ファームウェア	BMC ファームウェア	MP ファームウェア	DHCP ファームウェア
rx1600-2	1.10	2.33	E.02.29	適用外
rx1620-2	2.11	3.48	E.03.13	適用外
rx2600-2	2.31	1.52	E.02.29	適用外
rx2620-2	3.10	3.47	E.03.13	適用外
rx4640-8	2.13 3.11	2.35 3.47	E.02.29 E.03.13	1.10 1.10

新しい rx4640 サーバのうちのあるものは、表に示すバージョンのうち、大きいバージョン番号のファームウェアで出荷されます。このマニュアルが出版される時点では、既存のサーバをアップグレードするユーザ向けの最新のファームウェアは、小さいファームウェア番号のものであります。OpenVMS は、両方のファームウェアでテストしています。rx4640 サーバの新しいファームウェアがリリースされたら、番号の大きいファームウェアにアップデートすることをお勧めします。

サーバのファームウェア・バージョンを確認するには、次の例のように、EFI (Extensible Firmware Interface) のコマンド INFO FW を使用します。(EFI のアクセス方法および使用方法については、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください)。

```
Shell> INFO FW
FIRMWARE INFORMATION
  Firmware Revision: 2.13 [4412]          1
  PAL_A Revision: 7.31/5.37
  PAL_B Revision: 5.65
  HI Revision: 1.02
  SAL Spec Revision: 3.01
  SAL_A Revision: 2.00
  SAL_B Revision: 2.13
  EFI Spec Revision: 1.10
  EFI Intel Drop Revision: 14.61
  EFI Build Revision: 2.10
  POSSE Revision: 0.10
  ACPI Revision: 7.00
  BMC Revision: 2.35                      2
  IPMI Revision: 1.00
  SMBIOS Revision: 2.3.2a
  Management Processor Revision: E.02.29  3
```

- 1 システムのファームウェア・リビジョンは、2.13 です。
- 2 BMC のファームウェア・リビジョンは、2.35 です。
- 3 MP のファームウェア・リビジョンは、E.02.29 です。

HP Integrity rx4640 サーバには、アップグレード可能ファームウェアを備えた、Dual Hot Plug Controller (DHPC) ハードウェアが含まれています。DHPC ファームウェアの現在のバージョンを確認するには、次の例のように、EFI の INFO CHIPREV コマンドを使用します。ホット・プラグ・コントローラのバージョンが表示されます。0100 という表示は、Version 1.0 を意味します。0110 という表示は、Version 1.1 を意味します。

```
Shell> INFO CHIPREV
```

```
CHIP REVISION INFORMATION
```

Chip Type	Logical ID	Device ID	Chip Revision
Memory Controller	0	122b	0023
Root Bridge	0	1229	0023
Host Bridge	0000	122e	0032
Host Bridge	0001	122e	0032
Host Bridge	0002	122e	0032
Host Bridge	0004	122e	0032
HotPlug Controller	0	0	0110
Host Bridge	0005	122e	0032
HotPlug Controller	0	0	0110
Host Bridge	0006	122e	0032
Other Bridge	0	0	0002
Other Bridge	0	0	0008
Baseboard MC	0	0	0235

ファームウェアのアップグレード手順については、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

1.8.4 Fibre Channel ストレージ・デバイスからのブート

V8.2

多くのユーザは、速度の点と、SAN 内で共通クラスタ・システム・ディスクとしてサービスできるという点から、Fibre Channel (FC) ストレージ・デバイスからのブートを選びます。OpenVMS I64 システムでの FC ストレージ・デバイスからのブート方法は、OpenVMS Alpha システムでの FC ストレージ・デバイスからのブート方法と大きな違いがあります。

OpenVMS I64 システムでの FC デバイスの構成方法および FC デバイスからのブートについては、『OpenVMS Cluster 構成ガイド』の Fibre Channel の章を参照してください。

1.8.5 インストール DVD からのブート

V8.2

サポートする最小メモリ構成 (512MB) の I64 システムで、インストール DVD からブートすると次のようなメッセージが表示されます。

```
***** XFC-W-MemmgmtInit Misconfigure Detected *****  
XFC-E-MemMisconfigure MPW_HILIM + FREEGOAL > Physical Memory and no reserved memory for XFC  
XFC-I-RECONFIG Setting MPW$GL_HILIM to no more than 25% of physical memory  
XFC-I-RECONFIG Setting FREEGOAL to no more than 10% of physical memory  
***** XFC-W-MemMisconfigure AUTOGEN should be run to correct configuration *****
```

このメッセージは、システム・キャッシュ (XFC) の初期設定で、インストール中のキャッシュ処理を有効にするために、SYSGEN パラメータの MPW_HILIM と FREEGOAL を正常に調整したことを意味しています。インストールは続行できます。

1.8.6 I64 システムのリポート設定

V8.2

I64 システムは、リポートするようにコンソールから設定されていない限り、自動リブートは行なわれません。クラスタ環境では、Alpha システムはリポートされますが、I64 システムはシャットダウン後、コンソール・プロンプトの状態になります。

I64 システムがリポートするように設定する方法については、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

1.8.7 HP DECwindows Motif に関する注意事項

HP DECwindows Motif に関する以降の注意事項は、OpenVMS I64 ユーザを対象としています。

1.8.7.1 キーボードのサポート

V8.2

HP DECwindows Motif for OpenVMS I64 システムでサポートされているキーボード・モデルは、LK463 キーボードだけです。他のタイプのキーボードも OpenVMS I64 環境で動作する可能性がありますが、弊社では現在それらはサポートしていません。

1.8.7.2 サーバ起動前の周辺デバイスの接続

V8.2

システムを DECwindows X ディスプレイ・サーバとして適切に構成するには、起動前に、次のすべての周辺コンポーネントが接続されていなければなりません。

- モニタ
- USB マウス
- USB キーボード

接続されていない場合、サーバ・システムは初期化処理を正しく完了できないことがあります。たとえば、入力デバイス(マウスおよびキーボード)なしでサーバ・システムがスタートアップすると、ブランク・スクリーンになることがあります。

この問題を解決するには、すべての周辺コンポーネントを切り離してから再接続してください。

1.8.7.3 スタートアップ中に表示されるカウントダウン・メッセージ

V8.2

クライアント・オンリー・モード(サーバ無しの構成)で DECwindows Motif を実行すると、起動中に次のようなメッセージが表示されることがあります。

```
Waiting for mouse...  
Waiting for keyboard...
```

これらのメッセージは、デバイス・ポーリングの実行中だという情報を示しているにすぎません。15 秒経過すると、これらのメッセージは消えます。

このメッセージが表示されないようにするには、システムを起動する前に、入力デバイス(USB マウスおよび USB キーボード)を接続してください。

1.9 OpenVMS Alpha 向けの注意事項

以降のリリース・ノートは、OpenVMS Alpha ユーザ用です。

1.9.1 OpenVMS Alpha Version 8.2 のファームウェア

V8.2

OpenVMS Alpha Version 8.2 は、Alpha Systems Firmware CD-ROM Version 6.8 に収められているプラットフォーム固有ファームウェアでテストされています。ファームウェア CD-ROM に含まれなくなった古いプラットフォームについては、最新のファームウェア・バージョンで OpenVMS Alpha Version 8.2 がテストされています。

す。OpenVMS をアップグレードする前に、最新のファームウェアにアップグレードすることをお勧めします。

OpenVMS Alpha Version 8.2 キットには、Alpha Systems Firmware CD-ROM とリリース・ノートが同梱されています。ファームウェアをインストールする前にリリース・ノートに目を通してください。

Version 6.8. と、最新のファームウェアの情報については、次の Web サイトを参照してください (URL の大文字と小文字は区別されます)。

<http://ftp.digital.com/pub/Digital/Alpha/firmware/>

1.9.2 アップグレード・パス

V8.2

OpenVMS Alpha Version 8.2 では、下記のバージョンからのアップグレード・インストールをサポートしています。OpenVMS I64 Version 8.2 へのアップグレードはサポートしていません。

Version 7.3-2

Version 7.3-1

現在、OpenVMS Alpha Version 6.2x ~ 7.3 を実行している場合は、まず Version 7.3-1 または 7.3-2 にアップグレードした後、Version 8.2 にアップグレードする必要があります。なお、OpenVMS Alpha Version 7.3-1 システムの標準サポートは、2005 年 3 月 31 日で終了します。これ以降、OpenVMS Alpha V7.3-1 システムは、PVS (Prior Version Support) の対象とはなりません。OpenVMS オペレーティング・システムのサポートの詳細については、次の Web サイトの表を参照してください。

http://h71000.www7.hp.com/openvms/openvms_supportchart.html

これ以外のサポートされていないバージョンの OpenVMS を実行している場合は、以前のバージョンのドキュメントに記載されているアップグレード・パスに沿って、複数回のアップグレード作業を行わなければなりません。

クラスタの同時アップグレード

クラスタを一度にアップグレードする場合には、クラスタ全体をシャットダウンして、各システム・ディスクをアップグレードします。各コンピュータでアップグレードおよびリブートが完了するまでクラスタを使用することはできません。各コンピュータがリブートされると、それぞれのコンピュータではアップグレード後のオペレーティング・システムが実行されます。

クラスタのローリング・アップグレード

クラスタのローリング・アップグレードを実行する場合は、各システム・ディスクを個別にアップグレードします。この場合、新旧バージョンのオペレーティング・システムを同じクラスタ内で同時に実行できます(複合バージョン・クラスタ)。ただし、この場合システム・ディスクが複数必要です。アップグレード中でないメンバ・システムは使用可能です。

OpenVMS Alpha Version 8.2 を含む混合バージョン・クラスタでサポートされる OpenVMS Alpha と OpenVMS VAX のバージョンは、次のとおりです。

Version 7.3-2 (Alpha)

Version 7.3 (VAX)

クラスタ環境でアップグレードを行う場合、ローリング・アップグレードは OpenVMS Alpha オペレーティング・システム Version 7.3-2 および 7.3-1 からのアップグレードをサポートします。クラスタ内にこの他のバージョンが存在する場合は、それらのバージョンをサポートされているバージョンにアップグレードするまで、ローリング・アップグレードは実行できません。

どのバージョンのオペレーティング・システムも、V8.2 システムとの混合クラスタをサポートするには、修正キットをインストールする必要があります。詳細は、第 4.18.4 項を参照してください。

注意

現在 1 つのクラスタでは、アーキテクチャにかかわらず、一度に 2 つのバージョンの OpenVMS の実行のみがサポートされています。1 つの OpenVMS Cluster 内でサポートされるアーキテクチャは、2 つだけです。warranted サポートが提供されているのは、OpenVMS I64 Version 8.2 との組み合わせです。詳細は、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

OpenVMS オペレーティング・システムで warranted サポートが提供される組み合わせ、migration サポートが提供される組み合わせ、あるいは OpenVMS Alpha Version 8.2 のインストール/アップグレード手順、OpenVMS I64 Version 8.2 のインストール手順などの詳細は、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

1.9.3 Version 7.3-1 からのアップグレードでのエラー

V8.2

OpenVMS Alpha Version 7.3-1 のシステム・ディスクを Alpha Version 8.2 へアップグレードする際に、実行フェーズで次のメッセージが出力されることがあります。

OpenVMS ソフトウェアのインストールおよびアップグレードに関する注意事項

1.9 OpenVMS Alpha 向けの注意事項

```
%PCSI-E-PARUDF, file [SYSLIB]SYS$STARLET_C.TLB was not previously installed
or is present but out of scope; module update skipped
Terminating is strongly recommended. Do you want to terminate? [YES]
```

このエラー・メッセージは、状況を適切に説明していません。このメッセージでは、SYS\$STARLET_C.TLB ライブラリをアップデートできないとしています。これは正しくありません。このメッセージは、Alpha Version 7.3-1 システム・ディスクに適用したパッチ・キットが、ライブラリ内のモジュールを不適切にアップデートするために発生します。

アップデートを中止するようにとの推奨は無視してください。プロンプト Do you want to terminate? [YES] に対しては、NO を入力してください。アップデートは正常に続行されます。

1.9.4 Encryption for OpenVMS Alpha: 古いキットを削除する

V8.2

OpenVMS Alpha および Itanium システムの Encryption キットの製造社名が CPQ から HP に変更になりました。この名前は、PCSI ファイル名と PCSI データベースで見ることができます。たとえば、次のペアで旧名称と新名称とが解ります。

```
CPQ-AXPVMS-ENCRYPT-V0106-1.PCSI
HP-AXPVMS-ENCRYPT-V0106-1.PCSI
```

```
CPQ AXPVMS ENCRYPT V1.6
HP AXPVMS ENCRYPT V1.6
```

Alpha システムでは、ENCRYPT 製品が古いキット（つまり、Encryption 製品の製造社名として HP がリストされていなかったキット）からインストールされていると、アップデートの前に、まず ENCRYPT を削除する必要があります。次のコマンドを使用してください。

```
$ PRODUCT REMOVE ENCRYPT
$ PRODUCT INSTALL ENCRYPT
```

HP Encryption 製品を使用するのに、ライセンスは不要です。

1.9.5 HP DECram V3.n: アップグレード前に削除する

V8.2

OpenVMS Alpha Version 8.2 および OpenVMS I64 Version 8.2 以降、DECram は System Integrated Product (SIP) として、OpenVMS オペレーティング・システムとともに提供されます。OpenVMS Alpha システムを Version 8.2 へアップグレードする場合は、古いバージョンの DECram を削除しなければなりません。詳細は、

『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

DECram の注意事項については、第 2.13 節も参照してください。

1.9.6 アップグレード前の Kerberos V1.0 の削除

V7.3-2

POLYCENTER ソフトウェア・インストール・キットを使用して Kerberos Version 1.0 for OpenVMS をインストールした場合は、オペレーティング・システムをアップグレードする前に、POLYCENTER ソフトウェア・インストール・ユーティリティを使用して、Kerberos Version 1.0 を削除しなければなりません。(Version 2.0 を実行しているか、OpenVMS Version 7.3-1 オペレーティング・システムの一部として Version 1.0 がインストールされている場合は、Kerberos を削除する必要はありません。)

Kerberos を削除するには、インストール CD のメイン・メニューから、オプション 6 “Remove installed products”を選択します。削除する際に、データとディレクトリを削除したいか尋ねられます。(データとは、構成データ・ファイルと、(作成されている場合)主データベースを指します。)後で使用するためにこの情報を保存したい場合は、質問に対して“No”と応答します。メイン・メニューに戻り、OpenVMS のアップグレードを実行します。

アップグレード後、新しい Kerberos ディレクトリは、KRB\$ROOT:[*...]に配置されます。(KRB\$ROOT は、Kerberos が起動されるときに、システム・ワイドな論理名として定義されます。) Kerberos データは、構成時に作成されるか、古い Kerberos ディレクトリから移動されて、名前が変更されます。以前にインストールされていた Kerberos キットを削除し、データとディレクトリを保存した場合は、アップグレード後初めて Kerberos スタートアップ・プロシージャが実行されたときに、そのデータが自動的に新しいディレクトリに移動され、名前が変更されます。

次のコマンドを入力して、Kerberos サーバを起動します。

```
$ @sys$startup:krb$startup.com
```

Kerberos スタートアップ・プロシージャは、認識している Kerberos ファイルだけを移動し、名前を変更します。古い Kerberos ディレクトリ内にファイルを作成していた場合は、それらのファイルを手作業で移動しなければなりません。

Kerberos のインストールおよび構成の詳細は、『HP Open Source Security for OpenVMS, Volume 3: Kerberos』を参照してください。

1.9.7 アップグレード後に LANCP デバイス・データベースを変換する

V8.2

OpenVMS Alpha Version 8.2 へアップグレードするとき、アップグレード後の最初の LANACP の実行で LAN デバイス・データベースが自動的に Version 8.2 形式に変換されない場合は、この変換を実行する必要があります。

データベースを変換するには、次の LANACP コマンドを入力してデバイス・データベースを変換後 LANACP を停止させ、新しいデータベースで再起動できるようにします。

```
$ LANCP
LANCP> CONVERT DEVICE_DATABASE
LANCP> SET ACP/STOP
LANCP> EXIT
$ @SYS$STARTUP:LAN$STARTUP
```

1.9.8 DECnet-Plus の新しいバージョンが必要

V7.3-2

インストールまたはアップグレードを行って OpenVMS Alpha Version 7.3-2 以降のバージョンにするときには、DECnet-Plus の新しいバージョンもインストールしなければなりません。このインストールが必要な理由の 1 つに、AUTOGEN の動作の変更があります (第 4.3 節を参照)。

以前のバージョンの動作と異なり、DECnet-Plus for OpenVMS Version 7.3-2 以降では、AUTOGEN が必要としているように、製品情報を NEWPARAMS.DAT レコードに格納するようになりました。AUTOGEN は、DECnet-Plus にこの変更が行われていることを前提としているため、CLU\$PARAMS.DAT から「不正」レコードを削除する際に、警告を出力しません。AUTOGEN は、これらのレコードは古い DECnet-Plus キットが作成したものであるため、新しい DECnet-Plus キットが置き換えるものと見なします。このため通常の状態では、OpenVMS Version 7.3-2 以降のインストールまたはアップグレードの際に、動作に大きな違いは見られません。

ただし、他の製品が NEWPARAMS.DAT レコードに製品情報を格納しない場合、AUTOGEN は、レポートと、ユーザの SYS\$OUTPUT デバイスの両方に警告メッセージを出力します。これは、AUTOGEN がこの製品情報を必要とするようになったためです。この警告は、(製品名が付加されていないため) NEWPARAMS.DAT にあるパラメータ割り当てを AUTOGEN が受け入れられないことと、レコードが CLU\$PARAMS.DAT に追加されないことを示します。レコードが追加されないため、期待されている追加や変更が SYSGEN パラメータに行われず、リソースが足りなくなる可能性があります。ソフトウェア製品の開発者とテスト者は、この要件を意識しなければなりません。また、システム管理者にも関係する場合があります。

この新しい動作は、レイヤード・プロダクトのユーザと提供者の両方を守ることを目的としています。適切にアップデートされるようにこの情報を適切な順序にすると、不正なアップデートによって起こる問題を最小限にできます。

NEWPARAMS.DAT と CLU\$PARAMS.DAT の説明は、『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 の章を参照してください。

1.9.9 SHADOW_MAX_UNIT のデフォルト設定とメモリ使用量

V7.3-2

この注意事項は、以前の注意事項を更新するものです。以前の注意事項は、このシステム・パラメータのデフォルト設定について説明していましたが、デフォルト設定で消費されるメイン・メモリの量については説明していませんでした。

OpenVMS Alpha Version 7.3 で、HP Volume Shadowing for OpenVMS でのミニコピーがサポートされました。ミニコピー機能の一環として、新しいボリューム・シャドウイング・システム・パラメータ、SHADOW_MAX_UNIT が導入されました。OpenVMS Alpha システムでは、このシステム・パラメータのデフォルト値は 500 で、メイン・メモリを 24 KB 消費します。OpenVMS VAX システムでは、デフォルト値は 100 で、メイン・メモリを 5 KB 消費します。

Volume Shadowing for OpenVMS を使用する予定がない場合は、この設定を最小値の 10 に変更することができます（この場合、消費するメイン・メモリは 480 バイトです）。デフォルト値から最小値に変更すると、OpenVMS Alpha システムでは 23.5 KB のメイン・メモリ、VAX システムでは 4.5 KB のメイン・メモリを解放することができます。

注意

SHAD_MAX_UNIT は、静的なシステム・パラメータです。新しい設定値を有効にするには、システムをリブートしなければなりません。

ボリューム・シャドウイングの SHADOW_MAX_UNIT 設定の推奨値については、『Volume Shadowing for OpenVMS 説明書』を参照してください。

OpenVMS の関連製品に関する注意事項

この章では、OpenVMS 関連製品について説明します。関連製品のインストール/アップグレードに特に関連する事項については、第 1 章を参照してください。

コンパイラ、リンカ、ランタイム・ライブラリ・ルーチンの使用に関する注意事項については、第 5 章を参照してください。

2.1 関連製品のサポート

Software Public Rollout Reports for OpenVMS には、OpenVMS Alpha および OpenVMS VAX 用の Software Products Library キット (CD-ROM)、および OpenVMS I64 用の Layered Product Library キット (DVD) で提供されている弊社のソフトウェア製品の情報が記載されています。

レポートには製品名とバージョン、製品をサポートするのに必要なオペレーティング・システムのバージョン、製品のボリューム出荷日が示されています。レポートの情報は今後も追加され、変更される可能性があります。レポートは公開され、毎月更新されます。レポートの情報はたえず変化するので、リリース・ノートには含まれません。

Software Public Rollout Reports for OpenVMS は、次の Web サイトから入手できます。

<http://h71000.www7.hp.com/openvms/os/swroll/>

インターネットにアクセスできない場合は、四半期ごとに提供される Software Products Libraries からオペレーティング・システム・サポート情報を検索できます。このライブラリは、次のファイルにあります。

[README]SW_COMPAT_MATRIX.PS
[README]SW_COMPAT_MATRIX.TXT

また、Software Public Rollout Reports は弊社のサポート担当者から入手することもできます。

2.2 BASIC: STARLET ライブラリの作成に V1.5A が必要 (Alpha のみ)

V7.3-2

OpenVMS Version 7.3-2 およびそれ以降の変更により、V1.5A より前のバージョンの BASIC では、インストール中に BASIC\$STARLET ライブラリ・ファイルを作成できません。

STARLET ライブラリ・ファイルの構築を要求しなければ、以前のバージョンの BASIC を OpenVMS Version 7.3-2 およびそれ以降の上にインストールすることができます。また、以前にインストールされた BASIC コンパイラと、以前に作成された STARLET ライブラリ・ファイルは、古い OpenVMS システムを Version 7.3-2 およびそれ以降にアップグレードした後も動作します。

OpenVMS Version 7.3-2 およびそれ以降で動作しないのは、BASIC\$STARLET ライブラリ・ファイルの作成だけです。BASIC V1.5A キットには、STARLET ライブラリ・ファイルを OpenVMS Version 7.3-2 およびそれ以降上で正しく構築する拡張インストール・プロシージャが含まれています。

BASIC V1.5A は、最新のコンソリデーテッド・レイヤード・プロダクト CD-ROM で入手できます。

2.3 追加された CMAP ファイル

V8.2

次の新しい CMAP ファイルが、OpenVMS Version 8.2 国際化データ・キットで提供されています。

DECKANJI2000
GB18030
ISO8859-1-EURO
UTF8-20
UTF8-30

2.4 COBOL: I/O 実行時診断と RMS 特殊レジスタの変更

V7.3

OpenVMS Alpha Version 7.2 で Extended File Support が追加されたことにより、OpenVMS Alpha Version 7.2 以降での I/O 実行時診断メッセージおよび RMS 特殊レジスタの処理が変わりました。特に、Version 7.2 より前のバージョンの OpenVMS では、長いファイル名により RMS\$_FNM が発生しましたが、OpenVMS Alpha Version 7.2 以降のバージョンでは RMS\$_CRE が発生するようになりました。これら

の RMS に関する違いをサポートするために、新しい ODS-5 サポートを使用する必要はありません。

2.5 COM for HP OpenVMS (Alpha のみ)

ここでは、COM for HP OpenVMS に関する注意事項について説明します。

2.5.1 COM for OpenVMS のサポート

V8.2

現在 COM Version 1.4 for OpenVMS は、OpenVMS Alpha Version 7.3-2 および 8.2 でサポートされています。COM for OpenVMS についての最新情報は次の Web サイトを参照してください。

<http://h71000.www7.hp.com/openvms/products/dcom/>

2.5.2 高負荷時に発生するアプリケーションのレジストリ・アクセス・エラー

V7.3-2

CTLPAGES 値に 256 以下の値を設定して、COM for OpenVMS アプリケーションを負荷の高い状態で実行した場合、Error accessing registry database, contact system manager (0x000025fc) メッセージが出力されることがあります。この問題を回避するには、CTLPAGES 値に 512 を設定してください。

2.6 OpenVMS Version 8.2 では DECdfs Version 2.4 が必要

V8.2

OpenVMS Version 8.2 には、DECdfs Version 2.4 が必要です。古いバージョンの DECdfs を使用すると、エラー・メッセージが表示されます。

2.7 DECforms Web Connector Version 3.0 (Alpha のみ)

V7.3-1

DECforms がすでにインストールされている場合、DECforms Web Connector V3.0 を OpenVMS Version 7.3-1 以降で実行するには、次のタスクを実行します。

1. 次の行を削除するか、コメント・アウトします。

```
$ @SYS$COMMON:[JAVA$122.COM]JAVA$122_SETUP.COM
```

上記の行は、FORMS\$INSTALL_AREA ディレクトリの次のコマンド・プロシージャに含まれています。

- FORMS_SMGR_STARTUP.COM
 - FORMS_WEB\$STARTUP.COM
 - FORMS_WEB_CONFIG.COM
2. すべてのプロセスに対して、Java™環境がシステム全体に設定されているか確認します。システムの SYLOGIN.COM ファイルに Java 環境設定を追加することをお勧めします。
 3. ブラウザ・クライアントで Sun Java Plugin Version 1.2.2 が使用されているか確認してください(『SPD and the Administrative guide』を参照)。

2.8 DEC PL/I: OpenVMS での RTL のサポート

V7.3

OpenVMS オペレーティング・システムで配布される PL/I RTL と、Kednos Corporation が所有し配布する新しい PL/I RTL の間には、互換性はありません。OpenVMS オペレーティング・システムと同時に配布された旧バージョンが、より新しいバージョンを上書きしてしまうことがあります。該当するイメージ・ファイルは、SYS\$LIBRARY:DPLI\$RTL\$SHR.EXE です。

OpenVMS では、DCL の ANALYZE/IMAGE コマンドで次のように表示されるバージョンのファイルが配布されます。

Image Identification Information

```
image name: "DPLI$RTL$SHR"  
image file identification: "V4.0-6"
```

OpenVMS Version 7.3 以降にアップグレードする前に ANALYZE/IMAGE コマンドを実行し、より新しいバージョンの DPLI\$RTL\$SHR.EXE が存在する場合、コピーしておきアップグレード後に復元するか、後で PL/I キットを再インストールしてください。

第 5.30 節の、関連する注意を参照してください。

2.9 FMS キット

V8.2

次の FMS キットのいずれか(またはそれより新しいもの)を OpenVMS Alpha および OpenVMS I64 システムにインストールできます。

フルキット: HPFMS025

実行時キット: HPFMSRT025

2.10 Graphical Configuration Manager (GCM) (Alpha のみ)

Graphical Configuration Manager (GCM) は、オペレーティング・システムと一緒に出荷される Layered Products CD-ROM に含まれています。ただし、GCM は頻繁にアップデートされます。次の Web ページを定期的にチェックして、新しいバージョンのソフトウェアがリリースされていないか確認してください。

<http://h71000.www7.hp.com/openvms/products/gcm/>

2.11 HP DCE RPC for OpenVMS

ここでは、OpenVMS Version 8.2 の HP DCE RPC に関する注意事項について説明します。

2.11.1 DCE RPC による FAILSafe IP のサポート

V8.2

DCE RPC は FAILSafe IP の環境で動作するように拡張されました。

2.11.2 RPC ソケットにおけるバッファ・サイズ・チューニングのサポート

V8.2

DCE RPC は、論理名 `RPC_USE_DEFAULT_SOCKET_BUFFER_SIZE` を使用してソケットのバッファ・サイズをチューニングできるようになりました。この論理名を設定することで、RPC ランタイム・ライブラリで、システムのデフォルトのソケット・バッファ・サイズを使用できるようになります。論理名をシステム全体に対して定義するには、次のコマンドを実行します。

```
$ DEFINE/SYSTEM RPC_USE_DEFAULT_SOCKET_BUFFER_SIZE 1
```

RPC ランタイム・ライブラリの動作をオリジナルに戻すには、論理名 `RPC_USE_DEFAULT_SOCKET_BUFFER_SIZE` の割り当てを解除してください。

2.11.3 RTI (Remote Task Invocation) RPC (I64 のみ)

V8.2

I64 Version 8.2 の RPC は RTI RPC をサポートしていません。詳細については、キットに付属しているドキュメント『HP DCE for OpenVMS Alpha and OpenVMS I64 Release Notes』を参照してください。

2.11.4 Microsoft LAN Manager RPC はテストされていない (I64 のみ)

V8.2

NTLM (Microsoft 社の LAN Manager プロトコル) 上の Authenticated RPC は、OpenVMS I64 ではテストされていません。これは、DCE RPC が依存しているインフラストラクチャが、OpenVMS I64 上では使用できないためです。

2.11.5 utc_mulftime 係数引数の型

V8.2

DTSS API ルーチンの入力引数である係数 `utc_mulftime` は、I64 システムでは IEEE_FLOAT 型、Alpha システムでは G_FLOAT 型でなければなりません。CVT\$FTOF または CVT\$CONVERT_FLOAT を使用すると、`utc_mulftime` を呼び出す前に、この係数引数を適切な浮動小数点型に変換することができます。

2.11.6 G_FLOAT 浮動小数点型と IEEE 浮動小数点型のサポート

V8.2

DCE RPC for OpenVMS は現在、I64 プラットフォームと Alpha プラットフォーム上で、G_FLOAT 浮動小数点型と IEEE 浮動小数点型の両方をサポートしています。

1 つの RPC アプリケーション内では、同じ浮動小数点型を使用してください。異なる RPC アプリケーションであれば、1 つのシステム内でそれぞれ異なる浮動小数点型を使用しても適切に動作します。

I64 システム

I64 システム上では、DCE はデフォルトで IEEE_FLOAT 型を使用します。つまり、I64 システム用にビルドされた DCE アプリケーションは、IEEE_FLOAT 浮動小数点型を使用します。

C または C++ 言語で開発された RPC アプリケーションで G_FLOAT 浮動小数点型を使用するには、次の手順に従います。

1. RPC の実行時関数を呼び出す前に、新しい API 関数 `rpc_set_local_float_drep(RPC_APPLICATION_FLOAT_TYPE, &status)` を

呼び出します。RPC_APPLICATION_FLOAT_TYPE 定数に対して、コンパイラのコマンド行修飾子で指定された浮動小数点型が自動的に定義されます。詳細は、『Release Notes for OpenVMS DCE V3.2』を参照してください。

2. コンパイラの修飾子/FLOAT=G_FLOAT を使用して、RPC アプリケーション・プログラムをコンパイルします。
3. スタブのビルド時に、以下のうち適切な IDL コンパイル・オプションを使用します。

C アプリケーションの場合: -CC_CMD "CC/FLOAT=G_FLOAT"

C++ アプリケーションの場合: -CPP_CMD "CXX/FLOAT=G_FLOAT"

4. 以下のうち適切な DCE オプション・ファイルを使用して、RPC アプリケーションをリンクします。

C アプリケーションの場合: DCE.OPT

C++ アプリケーションの場合: DCE_CXX.OPT

C または C++ 言語で開発された RPC アプリケーションで IEEE_FLOAT 浮動小数点型を使用するには、次の手順に従います。

1. コンパイラの修飾子/FLOAT=IEEE_FLOAT (デフォルトのオプション) を使用して、RPC アプリケーション・プログラムをコンパイルします。
2. DCE.OPT または DCE_CXX.OPT を使用して RPC アプリケーションをリンクします。

Alpha システム

デフォルトでは、Alpha システム上でビルドされた DCE アプリケーションは、G_FLOAT 浮動小数点型を使用します。

C または C++ 言語で開発された RPC アプリケーションで IEEE_FLOAT 浮動小数点型を使用するには、次の手順に従います。

1. RPC の実行時関数を呼び出す前に、新しい API 関数 `rpc_set_local_float_drep(RPC_APPLICATION_FLOAT_TYPE, &status)` を呼び出します。RPC_APPLICATION_FLOAT_TYPE 定数に対して、コンパイラのコマンド行修飾子で指定された浮動小数点型が自動的に定義されます。詳細は、『Release Notes for OpenVMS DCE V3.2』を参照してください。
2. コンパイラの修飾子/FLOAT=IEEE_FLOAT を使用して、RPC アプリケーション・プログラムをコンパイルします。
3. スタブのビルド時に、以下のうち適切な IDL コンパイル・オプションを使用します。

C アプリケーションの場合: -CC_CMD "CC/FLOAT=IEEE_FLOAT"

C++ アプリケーションの場合: `-CPP_CMD "CXX/FLOAT=IEEE_FLOAT"`

4. 以下のうち適切な DCE オプション・ファイルを使用して、RPC アプリケーションをリンクします。

C アプリケーションの場合: `DCE.OPT`

C++ アプリケーションの場合: `DCE_CXX.OPT`

C または C++ 言語で開発された RPC アプリケーションで `G_FLOAT` 浮動小数点型を使用するには、次の手順に従います。

1. C または C++ コンパイラの修飾子 `/FLOAT=G_FLOAT` (デフォルトのオプション) を使用して、RPC アプリケーション・プログラムをコンパイルします。
2. `DCE.OPT` または `DCE_CXX.OPT` を使用して RPC アプリケーションをリンクします。

2.12 HP DECnet-Plus for OpenVMS: X.25 データ・リンクがサポートされていない (I64 のみ)

V8.2

HP X.25 for OpenVMS Alpha ソフトウェアは移植作業中で、まだサポートされていません。このため、HP DECnet-Plus for OpenVMS I64 Version 8.2 は、X.25 データ・リンクをサポートしていません。

2.13 HP DECram

ここでは、DECram に関する注意事項について説明します。

2.13.1 DECram は OpenVMS Version 8.2 に含まれる

V8.2

OpenVMS Alpha および OpenVMS I64 の Version 8.2 以降、DECram は System Integrated Product (SIP) として OpenVMS オペレーティング・システムに統合されて提供されます。ユーザは、DECram のライセンスを引き続き必要とします。DECram ドライバは、`SYS$COMMON:[SYS$LDR]` にあります。Alpha のユーザは、システム固有のディレクトリ (`[SYSx.SYS$LDR]`) 内に残っている `SYS$MDDRIVER` イメージを削除する必要があります。Version 8.2 へアップグレードする前に DECram の旧バージョンを削除する手順は、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

旧バージョンの DECram をロードしようとする、次のエラー・メッセージが表示されます。

```
SYSTEM-W-SYSVERDIF, system version mismatch; please relink
```

OpenVMS Version 8.2 では、旧バージョンの DECram はサポートされていません。

DECram Version 2.5 は、引き続き VAX システムでのみサポートされます。

2.13.2 DECRYPT DCL コマンドとの競合

V8.2

Encryption for OpenVMS Alpha レイヤード・プロダクトでは、インストール時に専用の DCL コマンド DECRYPT を生成します。DECRYPT は、DCL コマンド DECR のデフォルトの定義を上書きしますが、これはユーザが DECram を呼び出すために使用している可能性があります。

両方の製品がインストールされた場合、次のような DCL フォーリン・コマンド・シンボルを定義することにより DECram インタフェースにアクセスすることができます。

```
$ DECram == "$MDMANAGER"
```

2.13.3 DECram の最大ディスク・サイズ

V7.3-2

DECram のドキュメントでは、DECram の/SIZE と/CAPACITY に指定できる値の論理的な最大値は、利用可能な空きメモリの量に依存しますが、4,294,967,295 (つまり %XFFFFFFFF) と説明しています。

ただし、OpenVMS がサポートしている最大値は、ODS-2 ボリュームと ODS-5 ボリュームの場合、2,147,483,647 です。

2.13.4 DECram コマンドとエラー

V7.3-1

すべてのエラーがユーザ・インタフェースで返されるわけではないので、DECram コマンドの実行後は、ディスク・エラーのチェックを必ず実行してください。デバイスに固有のエラーは、システムのエラー・ログに送られます。DCL プロンプトで SHOW DEVICE MD と入力し、DECram コマンドの実行結果としてデバイスにエラーが発生していないか確認します。エラーを回復するには、エラー・ログ分析ツールを使用する必要があります。エラー・ログは、ASCII ファイル形式で記録されるた

め、SYS\$SYSROOT:[SYSERR]ERRLOG.SYS ファイルで MD-E-FAILURE という接頭辞を使用してエラーを探することができます。

2.13.5 DECram とボリューム・シャドウイング

V7.3-1

Volume Shadowing for OpenVMS を使用すると、DECram では、DECram ディスクを物理ディスクにシャドウできます。ただし、現在の Volume Shadowing for OpenVMS では、物理ディスクがなくなると、揮発性ディスクに書き込まれることに注意してください。Volume Shadowing for OpenVMS の今後のリリースでは、物理ディスクが存在しない場合に「書き込みを中止」する機能を組み込む予定です。

2.14 HP DECwindows Motif for OpenVMS

ここでは、HP DECwindows Motif for OpenVMS に関する注意事項について説明します。

2.14.1 DECwindows のポーズ画面でロック解除できない問題 (Alpha のみ)

V8.2

外部認証環境で、以下の条件が揃った場合は、ユーザは DECwindows のワークステーションのロックを解除できません。

- ユーザが SYSUAF データベース上で EXTAUTH としてマークされている。
- ユーザがユーザ名と/LOCAL_PASSWORD 修飾子を使用して DECwindows のワークステーションにログインしている。
- ポーズ画面が起動されている。

この状況下では、ワークステーションは有効なパスワードを認識できないため、パスワードの入力を要求し続けます。この状況から抜け出すには、別のソース（たとえばネットワークや LAT）からログインし、@SYS\$STARTUP:DECW\$STARTUP RESTART と入力して DECwindows サーバを再起動します。

この問題が将来のリリースで解決されるまでの間は、外部認証されたユーザが/LOCAL_PASSWORD 修飾子を使用して DECwindows のワークステーションにログインする必要がある場合は、ポーズ画面を呼び出さないように、自動ポーズ機能を無効にしてください。

2.14.2 新しいロケールの追加

V8.2

ローカライズされた DECwindows Motif ソフトウェアが使用する次の新しいロケールが、OpenVMS Version 8.2 国際化データ・キットに追加されました。

iw_IL.utf-8 (ヘブライ, イスラエル, UTF-8)
ko_KR.utf-8 (韓国, UTF-8)
zh_CN.utf-8 (中国, PRC, UTF-8)
zh_HK.utf-8 (中国, 香港, UTF-8)
zh_TW.utf-8 (中国, 台湾, UTF-8)

2.14.3 利用可能な言語バリエーション

V8.2

DECwindows Motif for OpenVMS Version 1.5 で用意されている言語は、日本語、Hanzi、およびハンゲル語です。ヘブライ語は今後用意される予定です。

DECwindows Motif の他の言語が必要な場合は、弊社のサポート担当に連絡して、その言語が用意されている以前のバージョンを確認するか、ソフトウェアの変換方法について相談してください。

2.14.4 LAT トランスポート・インタフェースのサポート

V8.2

DECwindows Motif Version 1.3 で廃止されていた、LAT トランスポート用の DECwindows Motif インタフェースのサポートが復活し、OpenVMS Alpha プラットフォームと OpenVMS I64 プラットフォームで利用できるようになりました。このサポートを使用すると、ユーザが、LAT X セッションを起動したり、DECwindows Motif Version 1.3-1 以降を実行しているシステムと低速のネットワークを介して通信できるようになります。また、DECwindows Motif システム上で動作するクライアント・アプリケーションが、LAT トランスポートを使用して X 端末システムに接続できるようになります。

OpenVMS オペレーティング・システムに含まれている、復活した LAT インタフェースは、DECwindows Motif Version 1.3-1 (またはそれ以降) と OpenVMS Version 7.3-2 (またはそれ以降) のディスプレイ・サーバと通信するための、有効なネットワーク・トランスポートとして使用できます。ただし、X11R6.6 環境の他の通信プロトコルとの使用はサポートされていません。これには、次の通信が含まれます。

- Inter-Client Exchange (ICE) および Session Manager プロトコル
- Low-Bandwidth X (LBX) プロキシ・サーバ

- プロキシ・マネージャ・アプリケーション
- フォント・サーバ

また弊社では、復活した LAT トランスポート・インタフェースによるトークン・ベースの認証プロトコル (MIT-MAGIC-COOKIE-1 や MIT-KERBEROS-5 など) の使用はサポートしていません。

2.14.5 ユーザが作成したトランスポートはサポートされない

V7.3-2

DECwindows Motif Version 1.3 for OpenVMS Alpha では、クライアント間交換 (ICE) プロトコル、低バンド幅 X (LBX) プロキシ・サーバ、および入力メソッド・サーバのマルチスレッドおよび通信のニーズに応じて、DECwindows Motif のトランスポート・ライブラリが大きく変更されました。この結果、弊社は、DECwindows Motif Version 1.3 以降を実行しているシステムでの、ユーザ作成ネットワーク・トランスポートのサポートを中止しました。

既存のトランスポート (DECNET, TCPIP, LAT, および LOCAL) はそのまま使用でき、問題なく機能しますが、更新されたトランスポート・インタフェースに基づくユーザ作成トランスポートの設計および実装に対するサポートは今後行われません。『VMS DECwindows Transport Manual』はアーカイブに入れられ、新しいライブラリは公開されません。

ユーザが作成したトランスポートを実装し、そのトランスポートを DECwindows Motif Version 1.5 以降の環境に移行する場合は、移行戦略の作成方法について、弊社のサポート担当者にお問い合わせください。

2.15 HP Secure Web Server のサポート

V8.2

OpenVMS Alpha Version 7.3-2 および OpenVMS Version 8.2 (Alpha および I64) は、Secure Web Server (SWS) Version 1.3-* をサポートする最後のリリースです。OpenVMS Alpha Version 7.3-2 は、SWS Version 2.0 をサポートする最後のリリースです。

SWS Version 1.3-* および SWS Version 2.0 の代替機能は、SWS Version 2.1 で、2005 年後半にリリースされる予定です。今後の SWS の新機能や機能拡張は、Apache 2.0.* オープン・ソース・コードをベースにした SWS 2.1 で提供されます。

OpenVMS Alpha Version 7.3-2 が PVS (Prior Version Support) の状態である限り、SWS Version 1.3-* および SWS Version 2.0 に対するサポートは継続します。また、OpenVMS Version 8.2 がサポートされている限り、SWS Version 1.3-* に対するサポ

ートは継続します。これらの SWS バージョンのサポートには、必要と判断された修正パッチ、セキュリティ・パッチが含まれます。

2.16 Pascal

OpenVMS I64 システムでは Pascal V5.9 が必要です。

ここでは、OpenVMS Alpha システム上の HP Pascal に関する注意事項について説明します。

2.16.1 STARLET ライブラリの作成には V5.8A (以降) が必要 (Alpha のみ)

V7.3-2

OpenVMS Version 7.3-2 の変更により、V5.8A より前のバージョンの Pascal では、インストール中に STARLET ライブラリ・ファイルを作成できません。

STARLET ライブラリ・ファイルを作成してインストールするかという質問に "NO" とこたえと、OpenVMS Version 7.3-2 以降上に、以前のバージョンの Pascal をインストールできます。また、以前にインストールされた Pascal コンパイラと、以前に作成された STARLET ライブラリ・ファイルは、古い OpenVMS システムを Version 7.3-2 以降にアップグレードした後も動作します。

OpenVMS Version 7.3-2 以降で動作しないのは、Pascal のインストールの、STARLET ライブラリ作成の部分だけです。Pascal V5.8A キットには、STARLET ライブラリ・ファイルを OpenVMS Version 7.3-2 以降上で正しく構築する拡張インストール・プロシージャが含まれています。

Pascal V5.8A は、最新のコンソリデーテッド・レイヤード・プロダクト CD-ROM で入手できます。

2.16.2 アップグレード後の HP Pascal のインストール (Alpha のみ)

V7.3

この注意事項は、すべてのバージョンの HP Pascal と、すべてのバージョンの OpenVMS Alpha オペレーティング・システムにあてはまります。

OpenVMS のアップグレード後、HP Pascal を再インストールして、新しいバージョンの STARLET.PAS ファイルと他の定義ファイルを作成し、アップグレード後のシステムと一致させなければなりません。

OpenVMS のアップグレード後に HP Pascal を再インストールしない場合でも、コンパイラは正しく起動しますが、STARLET.PAS ファイルと他の定義ファイルに、OpenVMS のアップグレードで提供される新しい定義や修正後の定義が含まれません。

2.17 I64 システムでの WEBES および SEA のサポート

V8.2

WEBased Enterprise Services (WEBES) (System Event Analyzer (SEA) for OpenVMS を含むツール・セットのキット) は、Alpha システム上で利用可能ですが、WEBES V4.4 I64 キットが 2005 年前半に出荷されると I64 システム上でも利用可能になります。このキットが出荷されると、四半期ごとにアップデートされ、OpenVMS にパッケージされる System Tools CD にも WEBES V4.4 が含まれます。最新版の WEBES は、次の WEBES ホームページから入手できます。

<http://www.hp.com/services/webes>

WEBES Version 4.3.3 以降は、OpenVMS Alpha Version 8.2 上で確認されサポートされるようになりました。

一般ユーザ向けの注意事項

この章では、OpenVMS オペレーティング・システムのすべてのユーザのための情報をまとめます。一般に使用するコマンドとユーティリティに関する情報が含まれています。

オペレーティング・システムのこのバージョンで提供される新機能については、『HP OpenVMS Version 8.2 新機能説明書』を参照してください。

3.1 OpenVMS Freeware CD-ROM

V8.2

OpenVMS Version 8.2 メディア・キットには、OpenVMS Freeware Version 7.0 CD-ROM が付属しています。Freeware CD-ROM には、アプリケーションの作成や、OpenVMS システムの使用あるいは管理のためのフリー・ソフトウェア・ツールとユーティリティが収録されています。

システムに Freeware CD-ROM をマウントし内容を表示するには、CD-ROM を CD-ROM ドライブに挿入し、次のコマンドを入力します。フリーウェアの詳細については、FREEWARE_README.TXT ファイルを参照してください。

```
$ MOUNT/OVERRIDE=IDENTIFICATION ddcu:  
$ TYPE ddcu:[FREEWARE]FREEWARE_README.TXT
```

この MOUNT コマンドでは、*ddcu*の箇所に、OpenVMS システム上の CD-ROM ドライブまたは DVD-ROM ドライブのデバイス名を指定します。

適切なデバイスをマウントしたら、DIRECTORY や COPY などの標準の DCL コマンドを使用して、キット・ディレクトリに直接アクセスできます。各ディスクの[FREEWARE]ディレクトリに、フリーウェアの要約を記述したテキスト・ファイル、その他の資料があります。

3.1.1 フリーウェア・メニューが利用できない (I64 のみ)

V8.2

OpenVMS Freeware V7.0 CD-ROM に含まれている[FREEWARE]FREEWARE.COM はフリーウェア・メニュー・システム・インタフェースですが、これは OpenVMS I64 システムでは動作しません。

このメニュー・システム・インタフェースは、OpenVMS Alpha および OpenVMS VAX システム上で動作します。

OpenVMS I64, OpenVMS Alpha, または OpenVMS VAX システムの、DIRECTORY や COPY などの DCL コマンドを使用して、Freeware V7.0 CD-ROM の内容に直接アクセスできます。これは、Freeware V7.0 ディストリビューションにアクセスする際の望ましい方法です。

フリーウェアと Freeware ディストリビューションについての情報は [FREEWARE]FREEWARE_README.TXT に収められています。このファイルは Freeware V7.0 ディストリビューションの各ボリュームに格納されており、内容は TYPE コマンドまたはテキスト・エディタで確認できます。

3.2 オンライン・ヘルプの変更

V8.2

OpenVMS Version 8.2 では、以前はヘルプ・トピック DCL_Tips (以前は HELP SPECIFY 下に存在) の下にあったトピックが、ヘルプのトップ・レベルからアクセス可能な、個別のヘルプ・トピックとなりました。新しいヘルプ・トピックは、次のとおりです。

- HELP Date_Time (時刻の仕様を含む)
- HELP Expression
- HELP Filespec
- HELP Integer
- HELP Privilege
- HELP Protection
- HELP String
- HELP Symbol
- HELP UIC

SET SERVER と SHOW SERVER のヘルプも、単純化され、より簡単にアクセスできるようになりました。SET SERVER ACME /CONFIGURE を除いて、技術的な内容は変更されていません。ただし、次の項目は、個別のヘルプ・トピックになりました。

- SET SERVER ACME_SERVER
- SET SERVER REGISTRY_SERVER
- SET SERVER SECURITY_SERVER
- SHOW SERVER ACME_SERVER
- SHOW SERVER REGISTRY_SERVER

3.3 DCL コマンド

ここでは、DCL コマンドに関する注意事項について説明します。

3.3.1 ANALYZE/ERROR_LOG コマンドは OpenVMS I64 に移植されない

V8.2

ANALYZE/ERROR_LOG コマンドは、OpenVMS I64 には移植されていません。この機能を OpenVMS I64 に移植する予定はありません。

オンライン・ヘルプには、誤って ANALYZE/ERROR_LOG コマンドが含まれていません。

3.3.2 CREATE/MAILBOX: 一時的な制約

V8.2

CREATE/MAILBOX/TEMPORARY には、現在 CMEXEC 特権が必要です。この制約は将来のリリースでは解除されます。

3.3.3 DIAGNOSE コマンドはサポートされない

V8.2

DIAGNOSE コマンドは、OpenVMS Version 8.2 ではサポートされません。詳細については、第 A.4 節および第 A.6 節を参照してください。

3.3.4 SHOW LICENSE /OE および/HIERARCHY には SYSLCK 特権が必要 (I64 のみ)

V8.2

SHOW LICENSE/OE および SHOW LICENSE/HIERARCHY コマンドには、現在 SYSLCK 特権が必要です。この制約は将来のリリースでは解除されます。

SYSLCK 特権は持たないが、運用中のデータベース環境の特定の製品をシステムで実行するライセンスがあるかどうかを確認したい場合は、次の例に示すように F\$LICENSE 関数を使用することができます。

```
$ WRITE SYS$OUTPUT F$LICENSE("RMSJNL")
TRUE
$
```

3.4 Open Source Tools CD-ROM での DECmigrate の提供について

V8.2

OpenVMS Migration Software for VAX to Alpha (DECmigrate) は、OpenVMS Version 8.2 インストレーション・キットに含まれている Open Source Tools CD では提供されていません。OpenVMS Version 7.3-2 用のメディアにはこのソフトウェア・キットが含まれていましたが、OpenVMS Version 8.2 ではテストが完了していません。以前のバージョンの OpenVMS 用のソフトウェアについては、次の Web サイトから引き続き利用できます。

<http://h71000.www7.hp.com/openvms/products/omsva/omsva.html>

OpenVMS Version 8.2 でのサポートが可能になれば、アップデート版が掲載されません。

3.5 iconv の修正

V8.2

次の修正が、iconv に対して行われました。

- 以前は、出力ファイルにバージョンの制限があった場合、iconv の CONVERT コマンドはファイルを変換しませんでした。この問題が解決されています。
- 以前は、DEC Hanyu 用の iconv コンバータでは、DTSCS (Digital Taiwan Supplemental Character Set) を変換できませんでした。OpenVMS Version 8.2 から、iconv コンバータ DECHANYU_EUCTW および EUCTW_DECHANYU は DTSCS 文字との間の変換ができるようになっています。

3.6 HP Secure Web Browser

ここでは、HP Secure Web Browser に関する注意事項について説明します。

3.6.1 必要メモリ量の増加

V7.3-1

OpenVMS ワークステーションで、Mozilla ベースの HP Secure Web Browser (SWB) を使用している場合は、128 MB 以上のメモリが必要です。ただし、処理を安定させるために、メモリを 256 MB にすることをお勧めします。

3.6.2 ODS-2 ディスク・ボリュームで発生するインストール・エラー (I64 のみ)

V8.2

OpenVMS I64 用 HP Secure Web Browser (CSWB) Version 1.4 を ODS-2 ディスク・ボリュームにインストールすると、次のように PCSI エラーとなります。

```
%PCSI-E-OPENIN, error opening
ODS2$DISK:[SYS0.SYSCOMMON.][CSWB.RES]SAMPLE^.UNIXPSFONTS.PROPERTIES;* as input
-RMS-E-FND, ACP file or directory lookup failed
-SYSTEM-W-BADFILEVER, bad file version number
%PCSI-E-OPFAILED, operation failed
```

Do you want to terminate?とプロンプトが出たら "NO"と答えることでインストールを継続できます。インストールは正常に継続できます。

代替手段として、ODS-5 ディスク・ボリュームに HP Secure Web Browser をインストールすることもできます。

3.7 TECO エディタは I64 システムではまだ利用できない

V8.2

TECO エディタは、OpenVMS I64 ではまだサポートされていません。このエディタは、OpenVMS I64 の将来のリリースで提供される予定です。

システム管理に関する注意事項

この章では、システムの保守と管理、性能の管理、ネットワーキングに関連する情報をまとめます。

このバージョンで提供される新機能の詳細については、『HP OpenVMS Version 8.2 新機能説明書』を参照してください。

4.1 システムのハングアップまたはクラッシュからの回復 (I64 のみ)

V8.2

システムがハングアップしたため、強制的にクラッシュさせたいときは、コンソールから Ctrl/P を押します。クラッシュ・ダンプの強制方法は、XDELTA がロードされているかどうかによって異なります。

XDELTA がロードされている場合、Ctrl/P を押すと、システムが XDELTA に入ります。システムは命令ポインタと、現在の命令を表示します。次の例のように、;C を入力することで XDELTA からクラッシュを強制できます。

```
$  
Console Brk at 8068AD40  
8068AD40!      add      r16 = r24, r16 ;; (New IPL = 3)  
;C
```

XDELTA がロードされていない場合、Ctrl/P を押すと、システムがプロンプト“Crash? (Y/N)”で応答します。Y を入力すると、システムがクラッシュします。これ以外の文字を入力すると、システムでは何も起こりません。

4.2 AUTHORIZE: DEFAULT と SYSTEM アカунトの新規クォータ

V8.2

DEFAULT と SYSTEM アカунトに関するクォータが更新されました。これらの更新されたクォータは、OpenVMS を新規にインストールした場合、または新しい SYSUAF データファイルを作成した場合に適用されます。既存の SYSUAF データ・ファイルは更新されません。

DEFAULT アカウントは、次のように更新されました。

クォータ	以前の値	新しい値
ASTLM	250	300
BYTLM	64,000	128,000
ENQLM	2,000	4,000
FILLM	100	128
PGFLQUOTA	50,000	256,000
TQELM	10	100
WSDEFAULT	2000	4,096
WSQUOTA	4000	8,192

SYSTEM アカウントの更新は、次の 2 つのクォータを除いて DEFAULT アカウントの場合と同じです。

クォータ	以前の値	新しい値
BYTLM	64,000	256,000
PGFLQUOTA	50,000	700,000

既存の SYSUAF ファイルが存在する前バージョンからアップグレードされたシステムでは、システム管理者が DEFAULT と SYSTEM アカウントのクォータを新しい値に更新しても構いません。

4.3 AUTOGEN: NEWPARAMS.DAT ファイルに関する新しい動作

V7.3-2

AUTOGEN は、レイヤード・プロダクト・キットが、製品名を含まない NEWPARAMS.DAT レコードを渡すことを許さなくなりました。これまでこの規則に従っていなかった、最も一般的に使用されている製品は、DECwindows と DECnet-Plus です。OpenVMS Alpha Version 7.3-2 またはそれ以降をインストールする際には、両方の製品の新しいバージョンをインストールしなければなりません (第 1.9.8 項を参照)。

AUTOGEN は、SYS\$SYSTEM:NEWPARAMS.DAT;* というファイルを探します。このファイルには、レイヤード・プロダクトが消費するシステム・リソースの量を示す、SYSGEN パラメータの変更が入っています。ソフトウェア・インストール・キットは、インストールされるソフトウェアの要件を満たすために、システム・マネージャに MODPARAMS.DAT を変更させる代わりに、NEWPARAMS.DAT ファイルを用意します。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 AUTOGEN の章を参照してください。

4.4 バックアップ・ユーティリティ: 動作変更

V8.2

物理的バックアップの復元に、同じ構造(トラック、シリンダ)の出力ディスクを必要としなくなりました。復元は、出力ディスクの容量が同じかそれ以上であれば機能します。

I64 システム・ディスクに対する BACKUP/IMAGE がサポートされます。I64 システム・ディスクのイメージは、Alpha システムと I64 システムのどちらでも保存と復元が可能です。

4.5 Oracle® 8iおよび9iでの DECdtm/XA (Alpha のみ)

V7.3-2

DECdtm/XA を使用して Oracle® 8i/9i XA 準拠リソース・マネージャ (RM) のトランザクションを調整する場合は、XA switch (xaoswd) の動的登録は使用しないでください。動的登録をサポートしている Version 9.0.1.0.0 の Oracle 共有ライブラリは、動作しません。必ず静的登録 XA switch (xaosw) を使用して、Oracle RM を DECdtm/XA Veneer にバインドしてください。

DECdtm/XA V2.1 Gateway は、クラスタ単位のトランザクション回復をサポートするようになりました。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、単一ノード障害から回復できるようになりました。残りのクラスタ・ノードで実行されているゲートウェイ・サーバは、障害の発生したノードの代わりに、トランザクション回復プロセスを起動できます。

4.6 デバイス・ユニットの最大数の増加

V8.2

以前のバージョンの OpenVMS では、10,000 個を超えるクローン・デバイス・ユニットを作成できませんでした。また、ユニット番号は、9999 の後は 0 に戻っていました。このことは、メールボックスや TCPIP ソケットなどの一部のデバイスにとっては、制限事項となっていました。

OpenVMS Version 7.3-2 からは、UCB\$L_DEVCHAR2 の DEV\$V_NNM ビットがクリアされており、DEVICE_NAMING システム・パラメータのビット 2 がクリアされている場合、OpenVMS は最大 32,767 個のデバイスを作成します。デバイス・ドライバの変更は必要ありませんが、最大のデバイス番号が 9999 であるという前提でコーディングされているプログラムやコマンド・プロシージャは、変更が必要になる場合があります。

4.7 ECP Data Collector と Performance Analyzer V5.5 (Alpha のみ)

V8.2

OpenVMS Alpha Version 8.2 用の Enterprise Capacity and Performance (ECP) Analyzer の推奨バージョンは、Version 5.5 です。Version 5.5 は、OpenVMS Version 6.2 以降のバージョンと下位互換性があります。

OpenVMS Version 8.2 では、ECP Collector は Performance Data Collector (TDC) Version 2.1 に置き換えられています。ECP Analyzer は、TDC Version 2.1 以降で生成された収集ファイルを解析できます。

OpenVMS I64 では、ECP Analyzer を現時点ではサポートしていません。

4.8 EDIT/FDL: 推奨バケット・サイズの変更

V7.3

OpenVMS Version 7.3 より前のバージョンでは、EDIT/FDL の実行時に計算されるバケット・サイズ(最大バケット・サイズは 63)が、常に最も近いディスク・クラスタのバウンダリに切り上げられていました。そのため、ディスク・クラスタ・サイズが大きい場合に、ファイルの元々のバケット・サイズは小さいが、バケット・サイズが必要以上に大きく切り上げられるという問題が発生することがありました。バケット・サイズが大きくなるほど、レコードとバケット・ロックの争奪が増加し、性能に大きく影響します。

OpenVMS Version 7.3 以降では、推奨バケット・サイズを計算するためのアルゴリズムが変更され、ディスク・クラスタが大きい場合に、より妥当なサイズが提案されます。

4.9 EFI\$CP ユーティリティ: 使用は推奨できない

V8.2

OpenVMS EFI\$CP ユーティリティは、現在ドキュメント化されておらず、サポートしないことになっています。このユーティリティは、OpenVMS Version 8.2 では使用しないでください。このユーティリティ内で行われている一部の特権操作により、OpenVMS I64 がブートできなくなることがあります。

4.10 EFI ツール: VMS_SHOW DUMP_DEV エラー (I64 のみ)

V8.2

DUMP_DEV リストが OpenVMS I64 Boot Manager ユーティリティ (BOOT_OPTIONS.COM) を使用して設定されている場合、EFI ユーティリティを使用して EFI シェルの DUMP_DEV EFI NVRAM 変数を表示すると、VMS_SHOW.EFI は、各 DUMP_DEV エントリに対して "Error: Unknown Device" を表示します。DUMP_DEV リストの表示には、OpenVMS I64 Boot Manager ユーティリティを使用してください。この問題は、次のリリースで解決する予定です。

4.11 Error Log Viewer (ELV) ユーティリティ: TRANSLATE/PAGE コマンド

V7.3-2

TRANSLATE コマンドで/PAGE 修飾子を使用してレポートを参照している際にメッセージが通知された場合、表示が乱れることがあります。この問題を回避するには、Ctrl/W を使用して、表示をリフレッシュします。

メッセージが通知された直後に Ctrl/Z を押すと、プログラムが突然終了します。この問題を回避するには、通知されたメッセージを越えてスクロールした後に Ctrl/Z を押します。

4.12 外部認証

ここでは、外部認証に関する注意事項について説明します。外部認証は OpenVMS Version 7.1 で導入されたオプションの機能であり、この機能を利用すると、OpenVMS システムは外部のユーザ ID とパスワードを使用して、指定されたユーザを認証できます。外部認証の使用方法についての詳細は、『OpenVMS システム・セキュリティ・ガイド』を参照してください。また、外部認証に関連するリリース・ノートについては、第 2.14.1 項を参照してください。

4.12.1 I64 外部認証サポート

V8.2

Advanced Server for OpenVMS V7.3A ECO4 (およびそれ以降) の製品キットには、OpenVMS Cluster 内の I64 システムに対するスタンドアロン外部認証ソフトウェアが含まれています。

I64 が動作している OpenVMS Cluster メンバのノードで NT LAN Manager の外部認証を可能にする場合は、Advanced Server がインストールされている Alpha システムから I64 メンバ・ノードへ I64 スタンドアロン外部認証イメージをコピーする必要があります。また、Advanced Server キットのリリース・ノートで説明されているように正しくセットアップされている必要があります。

4.12.2 DECterm 端末セッションでの SET PASSWORD の動作

V7.2

DECterm 端末セッションでは、ログインで使用する外部ユーザ名にアクセスすることができないため、SET PASSWORD 操作の際に外部ユーザ名を入力しなければなりません。外部ユーザ名のデフォルトは、プロセスの OpenVMS ユーザ名です。デフォルトが適切でない場合（つまり、外部ユーザ名と、それに対応する OpenVMS ユーザ名が異なる場合）、正しい外部ユーザ名を入力しなければなりません。

次の例に、外部ユーザ名が JOHN_DOE であるユーザが開始した SET PASSWORD 操作を示します。マッピングされた OpenVMS ユーザ名は JOHNDOE であり、これは SET PASSWORD 操作で使用されるデフォルトです。この場合、デフォルトは正しくないため、実際の外部ユーザ名がユーザによって指定されています。

```
$ set password
External user name not known; Specify one (Y/N) [Y]? Y
External user name [JOHNDOE]: JOHN_DOE
Old password:
New password:
Verification:
%SET-I-SNDEXTAUTH, Sending password request to external authenticator
%SET-I-TRYPWDSYNCH, Attempting password synchronization
$
```

4.12.3 ワークステーションではパスワードの有効期限切れは通知されない

V7.1

LAN Manager ドメインでは、パスワードの有効期限が切れると、ログインすることはできません。

PC のユーザには、外部ユーザ・パスワードの有効期限が間もなく切れることが通知されるので、有効期限が切れる前にパスワードを変更できます。ところが、外部認証を使用して OpenVMS ワークステーションからログインする場合、ログイン・プロセスは外部パスワードの有効期限が間もなく切れるかどうか判断できません。したがって、パスワードの有効期限が設定されていて、ユーザの大半が PC を使用していないサイトでは、ワークステーション・ユーザに対して外部認証を使用しない方が賢明です。

4.13 INITIALIZE コマンドの変更

V8.2

INITIALIZE コマンドのデフォルト動作は、次のように変更されました。

- 以前のバージョンの OpenVMS では、/ERASE 修飾子は、ボリューム特性 ERASE_ON_DELETE を設定し、ボリュームのイニシャル・データ・セキュリティ・イレース (DSE) も実行していました。Version 8.2 からは、新しいキーワードのどちらかまたは両方を /ERASE で指定することで、これらのアクションの 1 つまたは両方を個別に選択できます。

/ERASE=INIT DSE イレースを実行します。

/ERASE=DELETE ボリュームの ERASE_ON_DELETE フラグを設定します。

キーワードを指定しない場合の /ERASE のデフォルトの動作は、以前のリリースのままです (つまり、/ERASE は /ERASE=(INIT,DELETE) と同じです)。

- /CLUSTER_SIZE が指定されていないときの INITIALIZE のデフォルト動作が変更されました。ODS-2 と ODS-5 のいずれも、特定のサイズが設定されていないときには、デフォルトのクラスタ・サイズは最小の 16 となります (以前は 3 でした)。また、クラスタ・サイズが計算されるときには、値は 16 の倍数に切り上げられます。/CLUSTER_SIZE に値が指定されている場合の動作には変更はありません。

INITIALIZE コマンドのその他の変更点については、第 4.17 節を参照してください。INITIALIZE コマンドについての詳細は、オンライン・ヘルプ、または『OpenVMS DCL デictionary』を参照してください。

4.14 INSTALL ユーティリティ

V8.2

以前のバージョンで /TRACEBACK を指定してリンクされたプログラムは、/RESIDENT 修飾子を使用してインストールすることはできませんでしたが、この制限はなくなっています。/TRACEBACK を指定してリンクされたプログラムでも、そのプログラムが特権なしでインストールされる場合は、INSTALL で /RESIDENT 修飾子を使用できます。

4.15 ロック・マネージャ

ここでは、OpenVMS 分散ロック・マネージャに関する注意事項について説明します。

4.15.1 ロック値ブロックの拡張

V8.2

Alpha システムと I64 システムでは、ロック値ブロックが、16 バイトから 64 バイトに拡張されました。この拡張による利点については、次の資料を参照してください。

- 『OpenVMS System Services Reference Manual』: \$DEQ, \$ENQ, および \$GETLKI の説明を参照してください。
- 『OpenVMS Programming Concepts Manual』: 複合バージョン動作と相互運用性について参照してください。
- 『OpenVMS System Analysis Tools Manual』: SHOW RESOURCES 表示についての説明を参照してください。

アプリケーションでのこの機能を使用するかどうかは任意ですが、この機能を使用していなくても、各ロック・リソースを定義する RSB\$データ構造体のサイズが 48 バイトだけ大きくなります。

4.15.2 高速ロック再マスタリングと PE1

V7.3

ロック・マネージャには、ロック再マスタリングという機能があります。ロック再マスタリングとは、リソース・ツリーのロック・マスタの権利をクラスタ内にある別のノードに移動することです。ロック・ツリーのマスタになるノードは、クラスタ内の別のノードとのやり取りが不要なため、ローカルなロック要求をより高速で処理することができます。ほとんどのロック処理を実行するノードにロック・ツリーがあると、システム全体の性能が向上します。

OpenVMS Version 7.3 より前のバージョンで、ロック再マスタリングを実行すると、1つのローカル・ロックにつき1つのメッセージがすべてのノードから新しいマスタに送信されていました。このため、非常に大規模なロック・ツリーの場合には、ロック再マスタリング処理を実行するために膨大な時間が必要でした。しかも、この処理中には、ロック・ツリーに対するすべてのアプリケーションのロックが停止されました。

OpenVMS Version 7.3 以降では、ロック・データの新しいマスタへの送信は、非常に大規模な転送で実行されます。これはより効率的な処理であり、1つのロック・ツリーを 3 ~ 20 倍速く移動できるようになります。

ロック再マスタリングの大規模転送を使用することができるのは、OpenVMS Version 7.3 以降のバージョンを実行しているノードだけです。OpenVMS Version 7.3 以降のノードとそれより前のバージョンを実行しているノードとの間の再マスタリングでは、引き続き1つのロックにつき1つのメッセージを送信する必要があります。

PE1 システム・パラメータを使用して、再マスタリングの対象となりうるロック・ツリーのサイズを制限している場合には、その値を増やして大規模なロック・ツリーを移動できるようにするか、その値をゼロ (0) に設定してどのようなサイズのロック・ツリーでも移動できるようにします。

4.16 Logical Disk (LD) ユーティリティ:問題の修正

V7.3-2

以前のバージョンの OpenVMS では、Logical Disk (LD) ユーティリティが、コンテナ・ファイルのキャッシュをバイパスすることがありました。コンテナ・ファイルの読み取りや書き込みに RMS が使用された場合、ファイルへの接続と、その後の書き込み中の論理ディスクへの接続に LD ユーティリティが使用されると、RMS は古いデータを持つこととなります。

この問題は、LD ユーティリティを使用して、CD-ROM に書き込むイメージを作成する場合に、主に発生していました。

この問題は解決されたため、LD ユーティリティのコンテナ・ファイルとして使用されるファイルのキャッシュを、次の DCL コマンドを使用してオフにする必要はなくなりました。

```
$ SET FILE/CACHING_ATTRIBUTE=NO_CACHING CONTAINER_FILE.DSK.
```

4.17 On-Disk Structure (ODS) のレイアウトの変更

V8.2

OpenVMS Version 8.2 から、DCL の INITIALIZE コマンドに、新しい /GPT 修飾子が用意されました。この修飾子は I64 システム上でデフォルトで有効になり、GPT.SYS と呼ばれる新しいシステム・ファイルが作成されます。この新しい GPT.SYS ファイルには、OpenVMS のブートブロック情報が収められています (GPT は global unique identifier partition table の略です。)

Alpha システムでは、/NOGPT がデフォルトです。INITIALIZE コマンドの詳細は、オンライン・ヘルプ、または『OpenVMS DCL ディクショナリ』を参照してください。

GPT.SYS が存在する場合、このボリュームのボリューム論理ブロックのレイアウトは、Version 8.2 より前の Alpha システムでのレイアウトと異なります。GPT.SYS は、ボリューム上の最初と最後の論理ブロック (つまり、LBN 0 と LBN MAXBLOCK-1 [F\$GETDVI 関数に項目コード MAXBLOCK を指定して返される値に基づく]) を、最小サイズがそれぞれ 34 ブロックの 2 つのセグメントにマッピング

します。たとえば、/CLUSTER_SIZE=8 のボリュームでは、マップ領域は次のように表示されます。

```
DUMP/HEADER/BLOCK=COUNT=0 [000000]GPT.SYS
:
.
Map area
  Retrieval pointers
    Count:    40    LBN:      0
    Count:    40    LBN:  71132920
```

[000000]INDEXF.SYS ファイルの最初のブロックが LBN 0 に対応していることを前提としているプログラムを、GPS.SYS ファイルがある OpenVMS I64 システムまたは Alpha システム上で実行する場合、最初のブロック (VBN 1) の対応関係を変更しなければなりません。GPT.SYS では、VBN 1 はブート・ブロックには対応しなくなりました。INDEXF.SYS の LBN 0 と VBN 1 の間の対応関係に前提を置かず、INDEXF.SYS の VBN レイアウトを必ず使用するプログラムであれば、変更の必要はありません。ただし、このようになっていることを確認する必要があります。

INDEXF.SYS の VBN レイアウトについては、現在、『Guide to OpenVMS File Applications』の第 1.2.1 項と、Kirby McCoy 著『VMS File System Internals』(ISBN 1-55558-056-4, 1990) の第 2.5.1 項に記載されています。

注意

/GPT を指定すると、そのディスクは OpenVMS Version 7.2 よりも古いバージョンが動作しているシステムにはマウントできません。

INITIALIZE コマンドのその他の変更点については、第 4.13 節を参照してください。INITIALIZE コマンドについての詳細は、オンライン・ヘルプ、または『OpenVMS DCL デictionary』を参照してください。

4.18 OpenVMS Cluster システム

ここでは、OpenVMS Cluster システムに関する注意事項について説明します。

4.18.1 OpenVMS I64 でのクラスタのサポート

V8.2

一部の例外を除き、OpenVMS Cluster ソフトウェアは、OpenVMS Alpha および OpenVMS VAX システムでの機能と同じ機能を、OpenVMS I64 システムでも提供します。

4.18.2 現時点での制限事項

V8.2

一時的に、以下の例外があります。

- 以下に示すように、1つのクラスタ内に存在できるシステムの数、プラットフォームの構成に依存します。
 - 最大 8 台の I64 システム
 - 1つのアーキテクチャ混合クラスタ内で、Alpha システムと I64 システムの合計が最大 16 台 (I64 システムは 8 台を超えない)

1つのクラスタ内で 8 台を超える I64 システムのサポートは、2005 年の前半にアナウンスする予定です (総合計で 96 ノードまで)。

- サポートされる、I64 システムを含む運用クラスタには、VAX システムを含めることはできません。VAX システムは、開発や移行を目的とするクラスタに含めることはできますが、クラスタ内に VAX システムが存在するために問題が発生した場合は、VAX システムと I64 システムのどちらかを削除する必要があることを理解した上で使用する必要があります。詳細は、OpenVMS Cluster ソフトウェア SPD を参照してください。
- 現時点では、Open VMS Cluster システムの運用環境でサポートされるのは 2 つのアーキテクチャのみです。サポートされるクラスタ構成のリストは、『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。
- OpenVMS I64 システムのサテライト・ブートは、このリリースでは利用できません。今後のリリースでサポートされます。

4.18.3 制限事項

V8.2

OpenVMS Cluster ソフトウェアは、DSSI (DIGITAL Storage Systems Interconnect)、CI (Cluster Interconnect)、および Memory Channel という、3 つの独自仕様のインターコネクートを Alpha システム上でサポートしていますが、OpenVMS I64 システムではサポートしていません。DSSI ストレージと CI ストレージを直接 OpenVMS I64 システムに接続することはできませんが、Alpha システムに接続されている CI ディスクと DSSI ディスクは、同じクラスタ内の OpenVMS I64 システムから利用することができます。

SCSI クラスタと一般的に呼ばれる、SCSI インターコネクート上のマルチホスト共有ストレージは、サポートされていません (OpenVMS Alpha システムでも、新しい SCSI アダプタではサポートされていません)。ただし、業界標準の Fibre Channel 上のマルチホスト共有ストレージはサポートされています。

注意

OpenVMS Alpha システムにローカル接続されているストレージ (CI, DSSI, FC, または SCSI ストレージ) と, OpenVMS I64 システムにローカル接続されているストレージ (Fibre Channel または SCSI ストレージ) はどちらも, クラスタ内の他のメンバから利用できます。

4.18.4 クラスタの互換性のために必要なパッチ・キット

V8.2

OpenVMS Version 8.2 システムを既存の OpenVMS Cluster システムに導入する前に, 以前のバージョンの OpenVMS を実行しているシステムに, パッチ・キット (修正キットとも呼ばれます) を適用しなければなりません。Fibre Channel, XFC, または Volume Shadowing を使用している場合は, 追加のパッチ・キットも必要です。これらのキットは, 各バージョンに固有のものであります。

表 4-1 に示されているバージョンは, warranted サポート構成でも, migration サポート構成でもサポートされます。これらの構成についての詳細は, 『OpenVMS Cluster システム』または『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

表 4-1 に, パッチ・キットの適用が必要な機能と, そのパッチ ID 名を示します。各パッチ・キットには, 同じ名前の, 対応する readme ファイルがあります (ファイル拡張子は README です)。

次の Web サイトから, パッチ・キットをダウンロードしてください (「software patches」オプションの OpenVMS を選択します)。または, 弊社のサポート担当者に連絡して, ご使用のシステムに合ったメディアでパッチ・キットを入手してください。

<http://h18007.www1.hp.com/support/files/index.html>

注意

パッチ・キットは, 必要に応じて, 定期的にはアップデートされます。各機能に対する最新のパッチ・キット (キットの readme ファイルにバージョン番号が示されています) を必ず使用してください。各キットの最新バージョンは, Web サイトに掲載されているバージョンです。

表 4-1 クラスタの互換性のために必要なパッチ・キット

機能	パッチ ID
OpenVMS Alpha Version 7.3-2	
この項に示されているパッチ・キットを除く，すべてのパッチ・キットを持つアップデート・キット	VMS732_UPDATE-V0300
Fibre Channel/SCSI System	VMS732_FIBRE_SCSI-V0400 VMS732_SYS-V0600
OpenVMS VAX Version 7.3 ¹	
Audit Server	VAXAUDS01_073
Cluster	VAXSYSL01_073
DECnet-Plus	VAX_DNVOSIECO04-V73
DECwindows Motif	VAXDWMOTMUP01_073
DTS	VAXDTSS01_073
Files 11	VAXF11X02_073
MAIL	VAXMAIL01_073
MIME	VAXMIME01_073
MOUNT	VAXMOUN01_073
RMS	VAXRMS01_073
RPC	VAXRPC02_073
Volume Shadowing	VAXSHAD01_073
System	VAXSYS01_073

¹クラスタ内で VAX システムを使用する際の運用ガイドについては，第 4.18.2 項を参照してください。

VAX システムは I64 システムと同じクラスタに含めることはできませんのでご注意ください。クラスタ内で保証されている組み合わせの詳細については『HP OpenVMS Version 8.2 Upgrade and Installation Manual』を参照してください。

4.18.5 Fibre Channel および SCSI マルチパスと，他社製品との非互換性を修正する新しい API

V7.3-2

システムと，SCSI デバイスまたは Fibre Channel デバイスの間に存在する複数のパス間でのフェールオーバをサポートするマルチパス機能は，OpenVMS Alpha Version 7.2-1 で導入されました。OpenVMS Alpha Version 7.3-1 では，Fibre Channel マルチパス・テープ・デバイス間でのフェールオーバのサポートが導入されました。

このマルチパス機能は，他社のディスク・キャッシング，ディスク・シャドウイング，または類似の機能を持つ製品との互換性がないことがあります。この機能がソフトウェアの製造元でサポートされるようになるまでは，そのようなソフトウェアを，

マルチパス・フェールオーバー用に構成された SCSI デバイスまたは Fibre Channel デバイスでは使用しないでください。

OpenVMS Alpha SCSI ディスク・クラス・ドライバ (SYS\$DKDRIVER.EXE), OpenVMS Alpha SCSI テープ・クラス・ドライバ (SYS\$MKDRIVER.EXE), または SCSI 汎用クラス・ドライバ (SYS\$GKDRIVER) の Driver Dispatch Table (DDT) の変更依存している他社製品で SCSI マルチパス機能が正常に動作するようにするには、製品を変更する必要があります。

このようなソフトウェアの作成者は、OpenVMS Alpha Version 7.3-2 で導入された新しい DDT Intercept Establisher ルーチンを使用して、ソフトウェアを変更できるようになりました。これらのルーチンの詳細は、『HP OpenVMS Alpha Version 7.3-2 新機能説明書』を参照してください。

注意

他社製のディスク・キャッシュ製品や、ディスク・シャドウィング・アプリケーションを使用している場合は、アプリケーションがこれらの新しいルーチンを使用するように改訂されるまで、OpenVMS SCSI マルチパス構成や Fibre Channel マルチパス構成でこれらの製品を使用しないでください。

OpenVMS Alpha SCSI マルチパス機能と Fibre Channel マルチパス機能の詳細は、『OpenVMS Cluster 構成ガイド』を参照してください。

4.18.6 CLUSTER_CONFIG.COM と、ルート・ディレクトリ名の制限

V7.3-2

この注意事項は、『OpenVMS Cluster システム』の表 8-3 (「CLUSTER_CONFIG_LAN.COM および CLUSTER_CONFIG.COM から要求されるデータ」) のアップデートに関するものです。

このドキュメントでは、システム・ディスクに直接アクセスするコンピュータで使用できる、16 進数の桁数の制限を記載しています。この制限は、VAX コンピュータについては正しいのですが、Alpha コンピュータについては正しくありません。

このコマンド・プロシージャで、次の情報の入力が必要になります。

Computer's root directory name on cluster system disk:

このドキュメントには、次のように記載されています。

プロシージャから提供されるデフォルトをそのまま使用するときは、Return キーを押す。または SYSx という形式で名前を指定する。

- システム・ディスクに直接アクセスできるコンピュータの場合、xは1～9またはA～Dの16進数である(たとえば、SYS1やSYSA)。
- サテライトの場合、xは10～FFFFの値でなければならない。

システム・ディスクに直接アクセスできる16進値の範囲の制限は、VAXコンピュータについては正しく記載されています。システム・ディスクに直接アクセスできるAlphaコンピュータでは、この16進値の有効範囲はもっと広く、VAXの範囲と同じ1～9またはA～Dの他に、10～FFFFの範囲も含まれます。SYSEとSYSFは、システム用に予約されています。

『OpenVMS Cluster システム』の次の版には、この情報が盛り込まれます。

4.18.7 CI と LAN との間の回線切り替えによるクラスタの性能の低下

V7.3-1

CIと、複数のFDDI、100 Mb/s または Gb/s のイーサネット・ベースのCIRCUITの両方を含むOpenVMS Cluster構成では、SCS接続がCI回線とLAN回線の間を約1分単位で移動することがまれにあります。この頻繁な回線の切り替えが原因で、クラスタの性能が低下したり、シャドウ・セット・メンバのマウント確認が行われる場合があります。

PEdriverでは、数秒間継続しているLAN輻輳を検出し、対処することができます。LANパスでの遅延時間の大幅な増加やパケットの損失が検出されると、PEdriverはそのパスを使用しなくなります。パスの性能が回復したことが確認されると、そのパスを再度使用するようになります。

限界条件下では、LANパスにクラスタ・トラフィックで使用する負荷が追加されると、遅延やパケットの損失が容認できる限界を超える場合があります。クラスタの負荷が取り除かれると、パスの性能は再度使用できる状態まで回復できる場合があります。

LAN回線の負荷クラスに限界LANパスを割り当てると、その回線の負荷クラスが増加してCIの負荷クラス値140を超えて限界パスが対象となる場合(また、反対にLAN回線の負荷クラスが減少して140を下回り限界パスが除外される場合)に、SCS接続はCI回線とLAN回線の間を移動します。

LAN回線とCI回線間の接続の移動を確認するには、CONNECTIONクラスとCIRCUITSクラスを追加したSHOW CLUSTERを使用します。

回避方法

接続の移動が頻繁に行われている場合は、次のいずれかの回避方法を使用してください。

- SCACP または Availability Manager を使用して、使用する回線またはポートにより高い優先順位を割り当て、自動的な接続割り当てと移動を無効にします。

SCACP コマンドの例を次に示します。

```
$ MC SCACP
SCACP> SET PORT PNA0 /PRIORITY=2      ! This will cause circuits from local
                                        ! CI port PNA0 to be chosen over
                                        ! lower priority circuits.

SCACP> SET PORT PEA0 /PRIORITY=2      ! This will cause LAN circuits to be
                                        ! chosen over lower priority circuits.
```

- SCACP SHOW CHANNEL コマンドを使用して、使用の切り替えが行われているチャンネルを確認します。次に、SCACP を使用して、特定のチャンネルに目的のチャンネルよりも低い値を割り当てて、そのチャンネルを明示的に除外することもできます。たとえば、次のように指定します。

```
SCACP> SET CHANNEL LARRY /LOCAL=EWB/REMOTE=EWB /PRIORITY=-2
```

max, max-1の範囲内にある CHANNEL および LAN デバイスの優先順位値は等価とみなされます。つまり、この両方のデバイスに、最大優先順位値が指定されているものとみなされます。チャンネルまたは LAN デバイスを使用対象から外す場合は、優先順位値に 2 以上の差をつける必要があります。

4.18.8 OpenVMS Cluster システムでの Gigabit Ethernet スイッチの制限事項

永続的な制限事項

Gigabit Ethernet スイッチを介して Gigabit Ethernet ノードを OpenVMS Cluster システムに追加しようとする、スイッチの自動ネゴシエーションに関する設定が Gigabit Ethernet アダプタと異なる場合には、失敗します。スイッチで自動ネゴシエーション有効と設定されている場合は、アダプタも同様に設定する必要があります。また、スイッチで自動ネゴシエーションが無効と設定されている場合は、アダプタも同様に設定する必要があります。

多くの Gigabit Ethernet アダプタは、デフォルトで自動ネゴシエーションが有効になっています。例外は、Alpha システムの DEGXA で、EGn0_MODE コンソール環境変数が、要求された設定を保持しています。その設定はスイッチの設定と同じにする必要があります。

ノードを追加するときに、スイッチとアダプタの設定の不整合により失敗する場合には表示されるメッセージが、誤解を招く可能性があります。たとえば、CLUSTER_CONFIG.COM を使用してノードを追加し、ローカルのページ・ディスクとスワップ・ディスクをインストールするオプションを選択していると、ディスク・サービスの問題であるかのように見えます。CLUSTER_CONFIG.COM を実行しているノードは "waiting for node-name to boot" というメッセージを表示する一方で、ブート・ノードは "waiting to tune system" というメッセージを表示するためです。使用可能なディスクのリストは、ネットワーク・パスが失われているため、まったく表示され

ません。ネットワーク・パスが失われているのは、Gigabit アダプタとスイッチの間の自動ネゴシエーションのミスマッチが原因であることが伝わりません。

この問題を回避するには、次の手順で、新しいノードの Gigabit Ethernet アダプタの自動ネゴシエーションを無効にします。

1. 最初にノードをクラスタ内でブートするときには、会話型ブートを実行する。
2. 新しいノードのシステム・パラメータ LAN_FLAGS の値を 32 に設定して、システム内の Gigabit アダプタすべての自動ネゴシエーションを無効にする。

この初期構成を行った後、LAN_FLAGS システム・パラメータの、自動ネゴシエーションに関する設定を、システム内の Gigabit Ethernet アダプタすべてについて、スイッチ設定と一致させる必要があります。すべてのアダプタで自動ネゴシエーションを無効にする必要がない場合、LANCP デバイス・データベースに実行時設定を適切に設定する必要があります。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 LANCP の章を参照してください。

4.18.9 マルチパス・テープ・フェールオーバの制限事項

V7.3-1

Fibre Channel マルチパス・テープ・セット内の 1 つのデバイスで INITIALIZE コマンドを実行している間は、そのセットの別のメンバへマルチパス・フェールオーバを実行できません。別のマルチパス・テープ・デバイスが初期化されている間に、現在のパスで障害が発生した場合は、テープ・デバイスが機能しているパスへフェールオーバした後に、INITIALIZE コマンドを再試行してください。

この制限は、今後のリリースで無くなる予定です。

4.18.10 SCSI マルチパス媒体チェンジャでは自動フェールオーバは行われない

V7.3-1

Fibre と SCSI 間のテープ・ブリッジを使用して Fibre Channel に接続されている SCSI 媒体チェンジャ (テープ・ロボット) 向けの OpenVMS Alpha Version 7.3-1 以降には、パスの自動切り替えが実装されていません。そのようなデバイスに対しては複数のパスを構成できますが、別のパスに切り替える場合は、SET DEVICE /SWITCH コマンドを使用してパスの手動切り替えを使用する方法しかありません。

この制限は、今後のリリースで無くなる予定です。

4.19 OpenVMS Galaxy (Alpha のみ)

ここでは、OpenVMS Galaxy システムに関する注意事項について説明します。

OpenVMS Galaxy がサポートされているのは、OpenVMS Alpha システムだけです。

4.19.1 Galaxy の定義

V8.2

『OpenVMS Alpha パーティショニングおよび Galaxy ガイド』は、OpenVMS Version 8.2 ではアップデートされていません。ここでは、前後関係により異なる、Galaxy という言葉の定義を詳しく説明します。

表 4-2 Galaxy の定義

Galaxy が使われる 文脈	説明
ライセンス	1 台のコンピュータで OpenVMS の複数のインスタンスを作成および実行するために必要です。このライセンスがないと、1 台のコンピュータで実行できる OpenVMS のインスタンスは 1 つだけです。
システム・パラメータ	メモリ共用を設定します。GALAXY に 1 を設定すると、ハード・パーティションにそのパラメータが設定されている OpenVMS インスタンスで、ハード・パーティション内の複数のソフト・パーティション間でメモリを共用することを指定します (1 つのハード・パーティション内で 3 つ以上のソフト・パーティションを実行でき、そのすべてではメモリを共用したくない場合があります)。このパラメータでは、ノードで共用メモリを使用するかどうかだけを指定します。連携して動作する OpenVMS の複数のインスタンスを実行するためにこのパラメータを使用する必要はありません。メモリの共用のための設定は、コンソールで希望の構成ツリーを設定することで実現できます。GALAXY に 0 を設定すると、メモリは共用されません (デフォルト)。
ソフト・パーティション	CPU の移動、API の使用、メモリの共用などが可能なように、複数の OpenVMS インスタンスを 1 台のコンピュータ内で連携させて動作させることができるようにします。プラットフォームのパーティショニング機能により、リソースを複数のソフト・パーティションに分けることができ、それぞれのパーティションで OS のインスタンスを実行することができます。ソフト・パーティションは、その中で実行されている OS インスタンスが参照および使用することができる、リソースのサブセットです。

4.19.2 OpenVMS Graphical Configuration Manager

V8.2

AlphaServer ES47/ES80/GS1280 の Galaxy 構成で OpenVMS Graphical Configuration Manager (GCM) がサポートされるようになりました。以前は、Graphical Configuration Utility (GCU) だけがサポートされていました。

4.19.3 ES40 上の Galaxy: 非圧縮ダンプの制限事項

永続的な制限事項

AlphaServer ES40 Galaxy システムでは、インスタンス 1 のメモリが 4 GB (物理) 以上から始まっている場合、インスタンス 1 から raw (非圧縮) ダンプを書き出すことはできません。代わりに、圧縮ダンプを書き出さなければなりません。

4.19.4 ES40 上の Galaxy: Fast Path の無効化

V7.3-1

AlphaServer ES40 システムで Galaxy を使用する場合、インスタンス 1 で Fast Path を無効化する必要があります。そのためには、そのインスタンスで SYSGEN パラメータ FAST_PATH を 0 に設定します。

インスタンス 1 で Fast Path を無効化しないと、インスタンス 0 のリポート時にインスタンス 1 での入出力がハングします。この状態は、PCI バスをリセットし、インスタンス 1 をリポートするまで続きます。共有する SCSI または Fibre Channel がある場合、共有ノードでの入出力がハングし、これらのデバイスへのすべてのパスが無効になります。

4.20 OpenVMS Management Station

V8.2

OpenVMS I64 Version 8.2 と OpenVMS Alpha Version 8.2 での OpenVMS Management Station の推奨バージョンは、Version 3.2D です。ただし、OpenVMS Management Station は、OpenVMS Version 6.2 以降に対して下位互換性を維持しています。

OpenVMS Version 8.2 をインストールすると OpenVMS Management Station Version 3.2D がインストールされます。

4.21 OpenVMS Registry は Version 2 フォーマットのデータベースを壊すことがある

V7.3-2

キー・ツリーに揮発性のサブキーを 8 個以上作成して、スタンドアロン・システムやクラスタをリポートした場合、リポート後にサーバが起動すると、OpenVMS Registry サーバは、Version 2 フォーマットの Registry データベースを壊すことがあります。

この問題を回避するには、以下のいずれかを実行します。

- 揮発性のキーを使用しない。
- Version 1 フォーマットのデータベースを使用する。

Advanced Server for OpenVMS と COM for OpenVMS は、揮発性のキーを作成しません。

4.22 セキュリティ: DIRECTORY コマンド出力の変更

V7.3-2

OpenVMS Version 7.1 以降では、Advanced Server (PATHWORKS) アクセス制御エントリ (ACE) を含むファイルに対して DCL コマンド DIRECTORY/SECURITY または DIRECTORY/FULL を実行すると、Advanced Server ACE それぞれの 16 進表現は表示されなくなります。その代わりに、各ファイルに対して検出された Advanced Server ACE の総数が“Suppressed n PATHWORKS ACEs”というメッセージに要約されます。

抑制されている ACE を表示するには、SHOW SECURITY コマンドを使用します。これらの ACE を表示するには、SECURITY 特権が必要です。実際に、このコマンドは、Windows NT®のセキュリティ・ディスクリプタ情報を示す%x86 ACE を含む、OpenVMS ACE を表示します。Windows NT セキュリティ・ディスクリプタ情報は、Advanced Server に関連しています。

4.23 サーバ管理プロセス (SMHANDLER)

V7.3-2

サーバ管理プロセス SMHANDLER は、このプロセスをサポートしている Alpha システム上で自動的に起動するようになりました。システム管理者は、古いスタートアップ・ファイル SYS\$STARTUP:SYS\$SMHANDLER_STARTUP.COM への参照を、SYSTARTUP_VMS.COM や他のサイト固有のスタートアップ・ファイルから削除しなければなりません。この参照は、SYSTARTUP_VMS.TEMPLATE からは削除されています。

バックグラウンド: SMHANDLER について

一部の Alpha システムでは、切迫しているハードウェア障害をシステム・ファームウェアが報告し、対処するのを助けるために、サーバ管理プロセスが起動されます。障害の状況はさまざまですが、一般的には、過度の温度上昇、ファンの故障、または電源の故障があります。SMHANDLER は、警告状態を OPCOM へ報告することがあります。また、故障したシステムの電源をシステム・ファームウェアがオフにしようとした場合に、SMHANDLER は OpenVMS のシャットダウンを開始することもあ

ります。多くの場合、OpenVMS のシャットダウン制御を行った方が、突然システムの電源が落とされるよりも、破損の可能性が低くなります。

アップ状態の時間をできるだけ長くにするために、システム・マネージャは、POWEROFF システム・パラメータに 0 を設定することができます。これにより、障害が発生したシステム上で SMHANDLER が OpenVMS をシャットダウンするのを防止できます。ただし、システム・ファームウェアがシステムの電源をオフにすることは防止できません。

4.24 SYSGEN: セキュリティ監査の修正

V7.3-2

以前は、SYSGEN の監査やアラームを有効にしても、変更されたパラメータの情報が示されませんでした。OpenVMS Version 7.3-2 では、この問題が修正されました。監査やアラームに、変更されたパラメータ、その古い値、および新しい値の情報が含まれるようになりました。

4.25 SYSMAN ユーティリティ: DUMP_PRIORITY コマンドの変更

V8.2

SYSMAN DUMP_PRIORITY コマンドには、次の変更が行われています。

- DUMP_PRIORITY ADD, MODIFY, および REMOVE

新しい修飾子/[NO]INFORMATIONAL が追加され、情報メッセージの出力を制御できるようになりました(たとえば、コマンド・プロシージャ内で)。デフォルトは、/INFORMATIONAL です。

- DUMP_PRIORITY ADD および MODIFY

すでに存在するエントリを追加する場合、または DUMP_PRIORITY MODIFY で既存のエントリを変更する際に処理結果が既存のエントリと重複する場合には、メッセージ "SMI-I-SDPRDUPIGN, duplicate record creation ignored" が出力されます。(これまでは、メッセージ "SMI-F-SDPRNODUP, duplicate records not allowed" が出力されていました。) MODIFY の場合には、既存のレコードは削除されません。

- DUMP_PRIORITY REMOVE

System Dump Priority レジストリから、存在しないエントリを削除しようとすると、メッセージ "SMI-I-SDPRNOTREM, no record removed" が出力されるようになりました。(これまでは、メッセージ "SMI-F-SDPRNOTFOUND, system dump priority record not found" が出力されていました。このメッセージは、DUMP_PRIORITY MODIFY で、変更するエントリが見つからないときに出力されます。)

4.26 システム・パラメータ

ここでは、システム・パラメータに関する注意事項について説明します。

4.26.1 新しいシステム・パラメータ

V8.2

OpenVMS Version 8.2 で導入された新しいシステム・パラメータは、次のとおりです。

ERLBUFFERPAG_S2
ERRORLOGBUFF_S2
SCSI_ERROR_POLL
SHADOW_HBMM_RTC
SHADOW_REC_DLY
SHADOW_SITE_ID
SYSSER_LOGGING
TTY_DEFCHAR3
VHPT_SIZE (I64 のみ)

DEVICE_NAMING パラメータが以前のリリースから導入されていましたが、ドキュメントに記載されていませんでした。このリリースから記載されました。

すべてのシステム・パラメータの定義については、オンライン・ヘルプ、または『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。

4.26.2 廃止されたシステム・パラメータ

V8.2

次のシステム・パラメータは、OpenVMS Version 8.2 で「廃止」扱いになりました。

BJOBLIM
BOOT_STYLE
EXUSRSTK
LAMAPREGS
NISCS_LAN_OVRHD
NJOBLIM
QBUS_MULT_INTR
SA_APP
SBIERRENABLE
SCSCONNECT

SD_ALLOCLASS
TAILORED
UDABURSTRATE
VECTOR_MARGIN
VECTOR_PROC
XFMAXRATE

4.26.3 廃止されたシステム・パラメータの表示

V8.2

SYSGEN, SYSMAN, SDA, および SYSBOOT ユーティリティの SHOW コマンドは、次のいずれかを実行しない限り、廃止されたシステム・パラメータを表示しません。

- 新たに追加された/OBSOLETE 修飾子を SHOW コマンドに指定する。

たとえば、廃止されたシステム・パラメータをすべて表示するには、次のコマンドを入力します。

```
SYSGEN> SHOW/OBSOLETE
```

- 廃止されたパラメータの正確な名前を、SHOW *parameter-name* コマンドで指定する。

たとえば、次のコマンドは、指定されたパラメータが廃止されている場合は、OBSOLETE と表示します。

```
SYSGEN> SHOW parameter-name
```

4.26.4 システム・パラメータの変更

V8.2

OpenVMS Version 8.2 では、デフォルト値、最小値、最大値を最適なものにするために、多数のシステム・パラメータがアップデートされています。

- 32,767 個のプロセスをサポートします。

OpenVMS は、最大 32,767 個のプロセスをサポートするようになりました。この変更のために、次のシステム・パラメータの最大値が大きくなり、それぞれ 32765 になりました。

BALSETCNT
IJOBLIM
MAXPROCESSCNT

- 非ページング・ルックアサイド・リストのトリミングがオフになりました。

非ページング・ルックアサイド・リストは、デフォルトでオフになりました。これらのリストのトリミングを行うと、非ページング・プールの変数リストが断片化する可能性があります。場合によっては、変数プール・リストが大幅に断片化され、システムの性能が低下することがあります。次のシステム・パラメータのデフォルト値は 100 です。

NPAG_AGGRESSIVE

NPAG_GENTLE

- CTLPAGES の最小値が大きくなりました。
CTLPAGES の最小値が大きくなり、64 になりました。この値は、ミニマム・ブートをサポートできる値です。
- CLISYMTBL の値が大きくなりました。
CLISYMTBL の最小値と最大値の両方が大きくなり (それぞれ 64 と 8192)、DCL シンボルの追加作成あるいは大規模な DCL シンボルの作成をサポートできるようになりました。CLISYMTBL の新しい最小値では、ミニマム・ブートもサポートします。
- CHANNELCNT のデフォルト値が大きくなりました。
CHANNELCNT のデフォルト値と最小値が大きくなり、多数のファイルをオープンするアプリケーションが、不十分な CHANNELCNT 値のために失敗する可能性が減りました。新しい最小値は 64 で、新しいデフォルト値は 512 です。
- グローバル・セクションに関連するパラメータが大きくなりました。
グローバル・セクションに関連するシステム・クォータが大きくなり、デフォルト値がより適切な値になりました。

GBLPAGFIL (デフォルト値: 16394)

GBLSECTIONS (デフォルト値: 1024)

- DUMPSTYLE のデフォルト値が、圧縮/選択型ダンプ用に変更されました。
DUMPSTYLE パラメータのデフォルト値が 9 に変更され、圧縮/選択型ダンプが生成されるようになりました。この変更により、システムのシステム・ダンプ・ファイルが小さくても、必要なデータがシステム・ダンプ・ファイル内に記録されるようになりました。
- QUANTUM のデフォルト値と最小値が小さくなりました。
高速なシステムでは、一定時間に CPU が実行できる処理量が増加しました。QUANTUM のデフォルト値を小さくして 5 にすると、CPU は 1 秒あたりに、より多くのプロセスをスケジューリングできます。最小値は、1 です。
- GH_*パラメータの値が大きくなりました。
GH_*パラメータの最大値が大きくなり、65536 (512 MB) になりました。これらのパラメータをすべて最大値にすると、システムはブートできなくなります。これらのパラメータは、上限が 2 GB の、32 ビットのアドレス空間を使用します。

I64 イメージには、共用可能な読み取り専用セグメントがあります。このセグメントをサポートするために、GH_RES_DATA パラメータのデフォルト値が、0 から 100 ページに拡大されました。

- WSMAX のデフォルト値が大きくなりました。

I64 システムで大規模な物理メモリをサポートするために、WSMAX のデフォルト値が大きくなり、131072 になりました。

- OpenVMS I64 で、PQL パラメータ値が大きくなりました。

PQL パラメータは、デタッチされたプロセスが作成されたときに、各種のクォータのデフォルト値と最小値を提供するために使用されます。これらの多くのパラメータの値が大きくなり、I64 システムでの大規模な物理メモリに対応できるデフォルト値と最小値となりました。新しいデフォルト値は、次のとおりです。

パラメータ	Alpha でのデフォルト値	I64 でのデフォルト値
PQL_DBYTLM	65536	262144
PQL_MBYTLM	2048	128000
PQL_DPGFLQUOTA	131072	700000
PQL_MPGFLQUOTA	4096	512000
PQL_DWSDEFAULT	2048	32767
PQL_MWSDEFAULT	1024	16384
PQL_DWSQUOTA	4096	65536
PQL_MWSQUOTA	2048	32768
PQL_DWSEXTENT	32767	131072
PQL_MWSEXTENT	4094	65536
PQL_DENQLM	2048	2048
PQL_MENQLM	64	64

- I64 でのページング・プールおよび非ページング・プールのデフォルト値が大きくなりました。

I64 でのいくつかのページングおよび非ページング・プール・パラメータのデフォルト値が、次のように大きくなりました。

パラメータ	デフォルト値
NPAGEDYN	4194304 (4 MB)
NPAGEVIR	16777216 (16 MB)
PAGEDYN	4194304 (4 MB)

- I64 でのシステム・ワーキング・セット・クォータのデフォルト値が大きくなりました。

SYSMWCNT のデフォルト値が大きくなり、8192 になりました。

- Alpha での KSTACKPAGES のデフォルト値が大きくなりました。

KSTACKPAGES のデフォルト値が大きくなり、1 ページから 2 ページに変更されました。この変更により、各種のデバイスを備えたさまざまなハードウェア・プラットフォームを、デフォルト値でブートできるようになりました。

これらのパラメータの詳細は、オンライン・ヘルプ、または『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。

4.26.5 MMG_CTLFLAGS: ドキュメントの誤り

V8.2

オンライン・ヘルプ、『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 Performance Management』でのシステム・パラメータ MMG_CTLFLAGS のビット 0 の記述が正しくありません。ビット 0 の正しい記述は、次のとおりです。

「このビットを設定すると、定期的に行われるが、その他のときにはアイドル状態となるプロセスからトリミングすることによる再生が有効になる。この状況が発生するのは、空きリストと変更リストの合計サイズが FREEGOAL の値の 2 倍より小さくなったときである。ビットがクリアされている場合は、この機能は無効である。」

『OpenVMS Performance Management』のビット 1 の記述にも誤りがあります。正しい記述は、次のとおりです。

「LONGWAIT 秒より長い時間アイドルであったプロセスを外部にスワップすることによる再生が有効になります。これは、空きリストのサイズが FREEGOAL の値より小さくなったときに発生します。」

4.27 Terminal Fallback Facility (TFF)

OpenVMS Alpha システムの Terminal Fallback Facility (TFF) には、フォールバック・ドライバ (SYS\$FBDRIVER.EXE)、共有イメージ (TFFSHR.EXE)、Terminal Fallback ユーティリティ (TFU.EXE)、フォールバック・テーブル・ライブラリ (TFF\$MASTER.DAT) が含まれます。

注意

ユーザが呼び出し可能なインタフェースとして公開されているものではないため、TFFSHR は IMAGELIB から削除されています。ただし、イメージは現

在でも SYS\$LIBRARY: ディレクトリにあります。

TFF を起動するには、次のように SYS\$MANAGER にある TFF スタートアップ・コマンド・プロシージャを起動します。

```
$ @SYS$MANAGER:TFF$SYSTARTUP.COM
```

フォールバックを有効にしたり、フォールバック属性を変更するには、次のように Terminal Fallback コーティリティ (TFU) を起動します。

```
$ RUN SYS$SYSTEM:TFU  
TFU>
```

端末に対するデフォルトのフォールバックを有効にするには、次の DCL コマンドを入力します。

```
$ SET TERMINAL/FALLBACK
```

OpenVMS Alpha TFF は、次の点で OpenVMS VAX TFF と異なります。

- Alpha システムでは、TFF フォールバック・ドライバの名前は SYS\$FBDRIVER.EXE です。VAX システムでは、TFF フォールバック・ドライバの名前は FBDRIVER.EXE です。
- Alpha システムでは、TFF は 16 ビット文字のフォールバックを処理できます。OpenVMS Alpha フォールバック・テーブル・ライブラリ (TFF\$MASTER.DAT) には、VAX ライブラリより 4 つ多い 16 ビット文字テーブルが含まれます。表 4-3 に、これらの追加テーブルを示します。

表 4-3 TFF 文字フォールバック・テーブル

テーブル名	ベース	説明
BIG5_HANYU	BIG5	BIG5 を CNS 11643 (SIGGCC) 端末/プリンタへ
HANYU_BIG5	CNS	CNS 11643 (SIGGCC) を BIG5 端末/プリンタへ
HANYU_TELEX	CNS	CNS 11643 を MITAC TELEX-CODE 端末へ
HANGUL_DS	KS	KS を DOOSAN 200 端末へ

これらのテーブルは主にアジア地域で使用されます。また、このテーブルの形式は、16 ビット文字フォールバックのサポートのために変更されています。

- Alpha システムでは、TFU コマンド SHOW STATISTICS はフォールバック・ドライバ (SYS\$FBDRIVER.EXE) のサイズを表示しません。

RT 端末は TFF ではサポートされません。

Terminal Fallback Facility の詳細については、OpenVMS ドキュメント Web サイトで、アーカイブ扱いになった『OpenVMS Terminal Fallback Utility Manual』を参照してください。

<http://www.hp.com/go/openvms/doc>

このマニュアルを参照するには、左側のサイドバーに表示されている“Archived documents”をクリックしてください。

4.28 タイム・ゾーンの変更

V8.2

OpenVMS Version 8.2 では、204 個のタイム・ゾーンの追加と既存のタイム・ゾーンのアップデートが行なわれ、タイム・ゾーンが合計 540 個になりました。このタイム・ゾーンのリストは、次の URL でアクセス可能なタイム・ゾーンのパブリック・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/>

『OpenVMS システム管理者マニュアル(下巻)』の付録に、すべてのタイム・ゾーンの一覧があります。

4.29 User Environment Test Package (UETP) (I64 のみ)

V8.2

User Environment Test Package (UETP) を使用する際には、以下の点に注意が必要です。

- ロード・フェーズ中に、UETMEMORY01 でアクセス・バイオレーションが時折発生します。これによって、実行が終了したり、実行結果の有効性に影響を与えることはありません。UETP は、それでも利用可能で、正しい結果を出力します。
- デバイス・フェーズは現在、アクセス・バイオレーションが発生して実行が完了しません。
- DECnet フェーズは問題なく動作します。クラスタ・フェーズは、まだテスト中です。このフェーズは問題なく動作しているように見えますが、何点か注意事項があります。また、他のシステムの名前が正しく出力されません。

4.30 推奨するキャッシュ方式

V8.2

Virtual I/O キャッシュ (VIOC) (別名 VAX Cluster キャッシュ (VCC)) は、OpenVMS I64 では利用できません。I64 システムでは、SYSGEN パラメータの VCC_FLAGS に 1 を設定しても、0 を設定したのと同じになります。したがって、キャッシュはロードされません。

Alpha システムと I64 システムの両方でお勧めできるキャッシュ方式は、拡張ファイル・キャッシュ (XFC) です。XFC の詳細は『OpenVMS システム管理者マニュアル』を参照してください。

OpenVMS Alpha の将来のリリースでは、VIOC はサポートされなくなります。

4.31 Volume Shadowing for OpenVMS

ここでは、HP Volume Shadowing for OpenVMS (ホスト・ベース・ボリューム・シャドウイング (HBVS) とも言います) に関する注意事項について説明します。

4.31.1 デバイス名の必要条件

V7.3-2

Volume Shadowing for OpenVMS は、完全デバイス名 \$allocclass\$ddcu: の ddc 部分が 3 文字のデバイス名をサポートしています。

このリリースより前では、完全デバイス名の ddc 部分が長いデバイス名 (たとえば、\$1\$DECRAM10:) を作成でき、これらのデバイスを正しくマウントできました。ただし、このようなデバイスをシャドウ・セットの一部としてマウントすると、操作上の問題 (たとえば、他のディスクをシャドウ・セットに追加した場合の %MOUNT-F-XSMBRS エラー) が発生します。

OpenVMS Alpha Version 7.3-2 からは、Mount ユーティリティは、デバイスを最初にマウントする際に、完全デバイス名の ddc 部分に対し 3 文字という必要条件を強制するようになりました。この必要条件を満たしていない名前のデバイスをマウントしようとする、次のエラー・メッセージが表示されます。

```
MOUNT-F-NOTSHDWDEV, not a valid shadow set member
```

4.31.2 DCL コマンド・プロシージャ内での SET SHADOW と SHOW SHADOW の使用についての警告

V7.3-2

新しい DCL コマンド SET SHADOW および SHOW SHADOW は、今後も進化します。将来のリリースでは、SHOW SHADOW/FULL 表示のデフォルトの表示および実装により、現在の表示フォーマットが変更されます。このため、DCL コマンド・プロシージャ内で現在の出力フォーマットを解析することでシャドウ・セットの情報を取得することは避けるようにお勧めします。SHOW SHADOW コマンドが表示する項目の大部分は、F\$GETDVI レキシカル関数を使用して取得することができます。

さらに、SET SHADOW コマンドの動作も変更されます。SET SHADOW を使用してシステム上のすべてのシャドウ・セットの特性を同時に設定する場合は、他の新しい修飾子に加え、新しい VALL 修飾子が必要です。

これらの新しいコマンドを使用する DCL コマンド・プロシージャを作成する場合は、これらの変更を念頭に置いてください。

4.31.3 書き込みビットマップと異種デバイス・シャドウイング (DDS) の注意事項

V7.3-2

Volume Shadowing for OpenVMS を使用するときには、書き込みビットマップと異種デバイス・シャドウイング (DDS) の間に相互作用があります。

DDS (OpenVMS Version 7.3-2 の新機能) を使用すると、異なるサイズのディスク・デバイスからなるシャドウ・セットを構築できます。(DDS についての詳細は、『HP OpenVMS Alpha Version 7.3-2 新機能説明書』と『Volume Shadowing for OpenVMS 説明書』を参照してください。)

書き込みビットマップは、完全コピーのオーバーヘッドなしでメンバを仮想ユニットに戻せるように、シャドウ・セットの仮想ユニットに対して行われたアプリケーションの書き込みを追跡します。ユーザがシャドウ・セット・メンバに対して DISMOUNT /POLICY=MINICOPY コマンドを実行した場合や、MOUNT/POLICY=MINICOPY コマンドを使用してシャドウ・セットをマウントした場合に、書き込みビットマップが作成されます。このビットマップが作成されるときサイズは、ボリュームの現在のサイズに依存します。

シャドウ・セットがマウントされるとき、そのシャドウ・セットの仮想ユニットの論理サイズは、最小のメンバ・ユニットのサイズになります。シャドウ・セットのメンバが削除された場合、仮想ユニットの論理サイズは、セット内に残っているメンバのサイズを基にして、再計算されます。その結果、仮想ユニットの論理サイズは、大きくなる場合があります。

シャドウ・セットに書き込みビットマップが作成されるとき、そのサイズは、シャドウ・セットの仮想ユニットの現在のサイズによって決まります。仮想ユニットのサイズが後で大きくなると、ビットマップは仮想ユニット全体をカバーできなくなります。その後、ビットマップを使用してミニコピー操作でシャドウ・セット・メンバを戻すと、仮想ユニット内でビットマップがカバーしていない部分は、フル・コピー操作でコピーされます。

この問題を、次の例で示します。

- シャドウ・セット DSA1: は、次の 3 つのメンバからなります。

\$1\$DGA20: (18 GB)

\$1\$DGA21: (36 GB)

\$1\$DGA22: (36 GB)

- 次のコマンドを使用して、ミニコピー・ビットマップ付きで、シャドウ・セットから \$1\$DGA22: を削除します。

```
$ DISMOUNT/POLICY=MINICOPY $1$DGA22:
```

書き込みビットマップのサイズは、シャドウ・セットの仮想ユニットの現在のサイズである、18 GB を基にして決められます。

- \$1\$DGA20: をシャドウ・セットから削除します。ファイル・システムで残りのメンバの 36 GB 全体を利用できるようにするには、次のコマンドを使用します。

```
$ SET VOLUME/SIZE DSA1
```

\$1\$DGA20 は、新しいボリューム・サイズよりも小さいため、このシャドウ・セットでは使用できなくなります。

- 次のコマンドを使用して、\$1\$DGA22: をシャドウ・セットに戻します。

```
$ MOUNT/SYSTEM DSA1:/SHADOW=$1$DGA22: label
```

DSA1: の論理サイズは 36 GB のままですが、ビットマップがカバーしているのは、最初の 18 GB だけです。

- \$1\$DGA22: の最初の 18 GB はビットマップを使用してミニコピーでコピーされ、残りの 18 GB は、フル・コピー操作でコピーされます。

小さいシャドウ・セット・メンバの削除を予定している場合は、ミニコピー・ビットマップ付きで大きなシャドウ・セット・メンバを削除する前に小さいメンバを削除すれば、大きなビットマップが作成され、短いビットマップで性能へ悪影響を及ぼすのを避けることができます。(上記の例では、\$1\$DGA22: を削除する前に \$1\$DGA20: を削除します。)

4.31.4 KZPDC (Smart Array 5300) の制限事項

V7.3-2

すべてのシャドウ・セット・メンバが、次のようなフォルト・トレラント・デバイスからなるデバイスを使用して形成されているという前提で、Volume Shadowing for OpenVMS は、KZPDC コントローラ (Smart Array 5300) とともに使用できます。

- RAID 1。コントローラ・ベース・ミラーリングとも呼ばれます。
- RAID 5。パリティ付きのストライピングです。
- RAID ADG (Advanced Data Guarding)。複数のパリティ・デバイスを用いたストライピングです。

KZPDC (Smart Array 5300) コントローラでのフォルト・トレラント・デバイスは、基礎となる LUN のいずれかでメディア障害が発生した場合に、データ・エラーを修復できるデバイスです。

OpenVMS Alpha Version 7.3-2 およびそれ以降は、総ブロック数が異なるメンバからなるシャドウ・セットをサポートしています。この新しい機能は、異種デバイス・シャドウイング (DDS) と言います。DDS を使用すると、KZPDC デバイスは、サポートされている任意のコントローラ下のデバイスとシャドウ化できます。

以前のすべてのバージョンの OpenVMS では、HBVS で複数メンバのシャドウ・セットを作成するためには、すべてのデバイスは、総ブロック数が同じでなければなりません。構成ユーティリティは、KZPDC デバイスや MSA1000 デバイス上のブロックの総数として、作成可能で、要求されたサイズに最も近い値を設定します。KZPDC と MSA1000 は同じ計算方法を使用するため、同じ要求サイズで両者に作成されたデバイスには、同じサイズが設定されます。これにより、HBVS で複数メンバのシャドウ・セットを作成できます。

注意

フォルト・トレラント・デバイスが使用されていない場合は、HBVS を使用して複数メンバのシャドウ・セットを作成できないこともあります。たとえば、単一メンバのシャドウ・セットが、1つのデバイス (物理ディスクまたは非フォルト・トレラント・デバイス) を使用して形成されたとします。その後このデバイスで回復不可能なデータ・エラーが発生した場合、HBVS を使用してこのシャドウ・セットに他のメンバを追加することはできません。2つ目のメンバがシャドウ・セットに追加されると、HBVS はソース・デバイス全体を読み取り、ターゲット・デバイスへコピーします。基本シャドウ・セット・メンバ (つまりソース・シャドウ・セット・メンバ) からデータ・エラーが読み取られると、HBVS は、現在のすべてのシャドウ・セット・メンバ (ソース・メンバとコピー・ターゲット) に、「パッド・スポット」を作成しようとしています。いずれかのシャドウ・セット・メンバで、パッド・スポットを作成する要求が失敗すると、シャドウ・セットは、1メンバに縮小されません。

4.31.5 シャドウ・セット・マージ遅延の算出方法の変更

V7.3-2

シャドウ・セットの補助なしマージ操作の実行中には、アプリケーションの読み込み I/O の性能が、次の 2 つの理由で低下します。

- すべての読み込み I/O で、データ一貫性チェックを行う必要がある。
- シャドウ・セット・マージ操作により、I/O 帯域幅の競合が発生する。

シャドウ・セット・マージ操作では、アプリケーションへのマージ I/O の影響を少なくするために、絞り込みメカニズムを採用しています。マージ処理は、システムの負荷が検出されたときにマージ I/O 操作間に遅延を入れることにより、絞り込まれます。この遅延を算出するロジックが、OpenVMS Alpha Version 7.3-2 用に再設計されました。新しいマージ遅延の計算では、デフォルトのパラメータ設定で、HSG80 などの一部の I/O コントローラでのマージ速度が速くなります。詳細は、『Volume Shadowing for OpenVMS 説明書』を参照してください。

4.31.6 ANALYZE/DISK/SHADOW コマンドの動作

V7.3-2

ANALYZE/DISK_STRUCTURE コマンドに/SHADOW 修飾子 (OpenVMS Version 7.3-2 での新修飾子) を指定すると、シャドウ・セット全体の内容、またはシャドウ・セット内の指定された範囲のブロックが、矛盾していないかチェックされます。

何らかの理由でシャドウ・セットのメンバに接続上の問題が発生すると、ANALYZE/DISK_STRUCTURE コマンドは受け取ったエラーを表示してから、DCL プロンプトに戻ります。

接続上の問題を解消し、同じシャドウ・セット上でこのユーティリティを再度実行するには、ANALYZE/DISK/SHADOW コマンドを再実行する前に、仮想ユニット上に一時ファイルを作成する必要があります。

また、作成後一度もフル・マージが行われていないシャドウ・セットの場合、このユーティリティは、シャドウ・セット・メンバ間の、説明可能な矛盾について報告することがあります。このような矛盾は、/ERASE 修飾子なしで DCL コマンド INITIALIZE/SHADOW を使用してシャドウ・セットが作成され、各ディスク・デバイスが異なる内容を持っていた場合に発生します。この矛盾はディスク破壊ではないということを認識することが重要です。異なると報告されたブロックは、まだ書き込みが行われておらず、古いデータが入っている可能性があります。矛盾が報告されたブロックも、ファイルに割り当てられていることがあります。これは、ファイルのエンド・オブ・データの位置と、割り当てられたスペースの終わりとの間に、書き込まれていないスペースが存在することがあるためです。

フル・マージを行うと、このような矛盾をなくすことができます。フル・マージを開始するには、DCL コマンド SET SHADOW/DEMAND_MERGE DSAxxx を実行します。コントローラ・ベースのミニマージをサポートしているコントローラ (HSJ50 など) がサービスを行っているデバイスの場合、このコマンドは、シャドウ・セットがクラスタ内の 1 つのノードにのみマウントされている間に実行しなければなりません。そうしないと、ミニマージが発生し、矛盾は解決できなくなります。単一メンバのシャドウ・セットにメンバを追加する場合は、フル・コピー操作を行うことでも、ファイル・システムの内部と外部の両方でディスクの一貫性が保たれます。

フル・マージ実行後に ANALYZE/DISK/SHADOW コマンドがエラーを報告した場合は、これらのエラーについて調査する必要があります。

ANALYZE/DISK/SHADOW コマンドの動作については、第 4.31.7 項も参照してください。

4.31.7 異種デバイス・シャドウ・セットでの ANALYZE/DISK/SHADOW コマンドの動作

V7.3-2

ANALYZE/DISK/SHADOW コマンドは、新しいメンバの追加後にシャドウ・セットが論理的に拡張されてからフル・マージが発生していない場合にも、説明可能な矛盾を報告することがあります。この問題を、次の例で説明します。

- シャドウ・セット DSA1: は、次の 2 つのメンバからなります。
 \$1\$DGA20: (18 GB)
 \$1\$DGA21: (36 GB)
- 2 つ目の 36 GB のメンバ \$1\$DGA22: を、フル・コピー操作でシャドウ・セットに追加します。
- コピーの完了後、\$1\$DGA20: をシャドウ・セットから削除します。
- この時点で、SET VOLUME/SIZE DSA1: コマンドを実行すると、シャドウ・セットの仮想ユニット DSA1: は、36 GB に拡張されます。そして、ANALYZE/DISK/SHADOW は、シャドウ・セットの内容の最初の 18 GB だけが \$1\$DGA22: にコピーされたため、矛盾を報告します。

ANALYZE/DISK/SHADOW が報告する矛盾は、問題となっているスペースにアプリケーションがまだ書き込んでいないので、無害です。

ANALYZE/DISK/SHADOW コマンドの動作については、第 4.31.6 項も参照してください。

4.31.8 /MINICOPY を使用したシャドウ・セット・メンバのディスマウント

V7.3

シングル・サイトまたはマルチ・サイトの OpenVMS Cluster 構成では、クライアント・システムでシャドウ・セットのメンバのディスマウントに、/MINICOPY 修飾子を使用して DISMOUNT コマンドを実行すると、DISMOUNT コマンドが失敗することがあります。

回避方法

最初の DISMOUNT コマンドが失敗した場合、次の例のようにコマンドを繰り返します。

```
$! The following commands are NOT executed on the WILD3 system.
$
$ SHOW DEVICE DSA5555
Device          Device          Error   Volume      Free  Trans Mnt
Name            Status          Count   Label       Blocks Count Cnt
DSA5555:        Mounted         0      $80$DKA107: 7994646   1  18
$80$DKA107:    (WILD3) ShadowSetMember  0 (member of DSA5555:)
$80$DKA302:    (WILD3) ShadowSetMember  0 (member of DSA5555:)
$80$DKA303:    (WILD3) ShadowSetMember  0 (member of DSA5555:)
$
$
$ DISMOUNT/POLICY=MINICOPY $80$DKA302:
%DISM-W-CANNOTDMT, $80$DKA302: cannot be dismounted
%DISM-F-SRCMEM, only source member of shadow set cannot be dismounted
$
$
$ DISMOUNT/POLICY=MINICOPY $80$DKA302:
$
```

この問題は、今後のリリースで修正される予定です。

プログラミングに関する注意事項

この章では、OpenVMS システムでのアプリケーション・プログラミングとシステム・プログラミングに関する注意事項について説明します。

5.1 プログラムの再コンパイルとリンクが必要 (I64 のみ)

V8.2

OpenVMS I64 Version 8.2 では、すべてのプログラムを再コンパイルしなければなりません。OpenVMS I64 の以前の評価リリースや、フィールド・テスト・リリースでコンパイルおよびリンクしている場合でも、再コンパイルが必要です。(弊社は、評価リリースに対して互換性のない変更を行う権利を有しています。)この OpenVMS I64 Version 8.2 の製品版リリースから、上位互換性を保つ弊社の通常の方針が適用されます。

Alpha プログラムの再コンパイルについては、第 5.2 節を参照してください。

5.2 特権プログラムの再コンパイルが必要 (Alpha のみ)

V8.2

メジャー・バージョン・リリースである OpenVMS Alpha Version 8.2 では、多くの特権データ構造体に変更されています。このため OpenVMS の内部データ構造体やルーチンを参照する/SYSEXEC でリンクされた特権アプリケーションは、再コンパイルおよび再リンクを行う必要があります。

イメージの起動時やデバイス・ドライバのロード時に SYSVERDIF エラー・メッセージが出力された場合は、特権イメージや特権ドライバが、以前のバージョンのオペレーティング・システムでコンパイルおよびリンクされていることを示しています。このイメージやドライバを OpenVMS Alpha Version 8.2 で実行するためには、再コンパイルおよび再リンクが必要になります。

I64 プログラムの再コンパイルについては、第 5.1 節を参照してください。

5.3 特権データ構造体の変更

V8.2

OpenVMS Version 8.2 では、多数の特権データ構造体が変更されています。これらの変更は、Alpha システムと I64 システムの両方で実施されています。これらのデータ構造体の変更の大半は、オペレーティング・システムでの将来のスケーリングおよび性能強化の計画を実現するためのものです。この変更の結果、ベース・オペレーティング・システムに対してリンクされているイメージやドライバ(つまり、LINK コマンドで/SYSEXEC を使用するもの)は、OpenVMS Version 8.2 で動作させるために再コンパイルと再リンクが必要となります。

特権データ構造体の変更は、すべての特権イメージやドライバに影響するとは限りません。変更されたサブシステムにリンクされているイメージやドライバだけが影響を受けます。変更されたサブシステムに関しては、サブシステムに対応するメジャー・バージョン識別番号が変更されています。変更されたサブシステムは、次のとおりです。

```
SYS$K_IO  
SYS$K_MEMORY_MANAGEMENT  
SYS$K_CLUSTERS_LOCKMGR  
SYS$K_FILES_VOLUMES  
SYS$K_CPU  
SYS$K_MULTI_PROCESSING  
SYS$K_PROCESS_SCHED
```

注意: I64 システムでは、これらのサブシステムは SYS\$K_VERSION_xxxx のように表示されます。

これらのサブシステムは、各種の特権システム・ルーチンおよびデータ・セルを使用しているイメージのリンク時に、そのサブシステムのバージョン番号がイメージに記録されています。ANALYZE/IMAGE ユーティリティを使用すると、特権イメージがどの特権サブシステムにリンクされているかを調べることができます。例を次に示します。

```
$ ANALYZE/IMAGE IMAGE.EXE /OUTPUT=IMAGE.TXT  
$ SEARCH IMAGE.TXT "SYS$K_"
```

変更されたサブシステムが表示された場合、OpenVMS Version 8.2 は、イメージの実行に失敗し、オペレーティング・システムの以前のバージョンにリンクされたイメージに対して SS\$_SYSVERDIF (システム・バージョン不一致エラー) を出力します。

5.3.1 KPB 拡張

V8.2

OpenVMS の以前のバージョンでは、IPL 2 より上のカーネル・モードでの KPB の使用をサポートしていました。I64 への移行を簡単にするために、KPB の使用が、アウター・モードとすべての IPL に拡張されました。この Alpha と I64 での変更により、Alpha と I64 の両方で、以前はプライベート・スレッド・パッケージを持っていた一部のコードが、KPB を使用できるようになりました。カーネル・プロセスでのこれらの変更をサポートするために、KPB 構造体に変更を行う必要がありました。既存の Alpha コードでは、ソース・プログラムを変更する必要はありません。

5.3.2 CPU の名前空間

V8.2

OpenVMS には現在、最大の CPU ID が 31 であるという、アーキテクチャ上の制限があります。各種の内部データ構造体およびデータ・セルでは、CPU マスクに 32 ビットを割り当てています。将来のリリースでより多くの CPU ID をサポートできるように、これらのマスクに割り当てられるスペースを、Alpha では 64 ビット、I64 では 1024 ビットに増やしました。既存のロングワードの CPU マスクのシンボルおよびデータ・セルは、今後も維持されます。

当面は、特権イメージおよびドライバには影響はありません。ただし、将来的には、CPU マスクを参照する特権付き製品が、31 個より多くの CPU ID を持つシステムをサポートするために、コードをどのように変更しなければならないかを明記する予定です。

5.3.3 64 ビットの論理ブロック番号 (LBN)

V8.2

OpenVMS はこれまで 31 ビットの LBN をサポートしています。このため、ディスク・ボリュームのサポートが 1 TB に制限されます。内部の LBN フィールドに割り当てられるスペースが 64 ビットに拡張されており、より大きなディスク・ボリュームを将来サポートできるようになります。既存のロングワードの LBN シンボルは今後も維持され、クォドワード・シンボルでオーバレイされます。

5.3.4 動的スピンロックのフォーク

V8.2

OpenVMS オペレーティング・システムを大規模な SMP システムへ拡張するために、オペレーティング・システム内の多くの部分で、限られた数の静的スピロックではなく、動的スピロックを使用するようになっていました。フォークする機能と、フォーク・ディスパッチャが動的ピンロックと同期をとる機能が必要とされます。FKB 構造体のサイズを拡張し、FKB\$L_SPINLOCK フィールドをこの構造体の最後に追加することによって、この機能を OpenVMS Version 8.2 に追加しています。このスピロック・フィールドは、FKB\$B_FLCK に値 SPL\$C_DYNAMIC が設定されている場合にのみ参照されます。FKB 構造体は、他の多くのシステム・データ構造体に埋め込まれているため、この変更は、多数の特権データ構造体のサイズとレイアウトに影響します。

FKB\$B_FLCK フィールドを、OpenVMS が作成した構造体から他の FKB へコピーするアプリケーションは、FKB\$L_SPINLOCK フィールド内のデータもコピーする必要があります。

FKB 構造体を割り当て、ハードコード化された値 32 をサイズとして使用していないか、特権コードをチェックしてください。コードでは、FKB のサイズとして、シンボル FKB\$C_LENGTH を使用しなければなりません。

5.3.5 UCB と DDB のアップデート

V8.2

UCB 構造体と DDB 構造体には、多数のアップデートが行われました。

DDB に対応する UCB のリストは、現在は単方向リストです。UCB を作成したり削除する場合は、適切な位置が見つかるまで、このリストをたどらなければなりません。OpenVMS Version 8.2 では、UCB は双方向リストで DDB にリンクされるようになりました。さらに DDB は、新しいユニットを作成するときに検索を開始する位置を示すシード・ポインタを維持しており、デバイスの作成が速くなります。プレート UCB 内でユニット・シード・ポインタを操作しているドライバには、デバイスの作成が速くなるという利点はありません。

DDB の UCB リストを操作するコードは、正しく動作しなくなります。UCB のリンクとリンク解除には、提供されている内部ルーチンを使用してください。UCB リストを前方にたどるコードは、引き続き正しく動作します。

UCB\$W_UNIT フィールドは現在、16 ビット・ワード・フィールドです。このフィールドには、32 ビットが割り当てられるようになりました。UCB\$W_UNIT フィールドは引き続き維持されるため、ソース・コードの変更は必要ありません。今後のリリースでは、OpenVMS はより大きいユニット番号をサポートする可能性があります。この機能をサポートできるドライバに対してだけ、この変更が行われる予定です。

ターミナル・ドライバの UCB 拡張内のバイト・フィールドとワード・フィールドは、ロングワード境界に揃えられるようになりました。

5.3.6 PCB\$T_TERMINAL のサイズの拡張

V8.2

Process Control Block (PCB) 構造体には、PCB\$T_TERMINAL というフィールドが含まれています。このフィールドは 8 バイトで、会話型プロセスのデバイス名 (LTA123:、RTA7:、NVA456: など) を保持します。このフィールドは、文字数付きの ASCII 文字列で、1 バイト目が文字列の長さ、残りの 7 バイトがデバイス名です。デバイス名が 3 文字の場合、ユニット番号には 4 桁しか使用できないため、999 より大きいユニット番号ではコロンは取り除かれます。OpenVMS Version 8.2 では、このフィールドは 16 バイトに拡張されたため、より大きなユニット番号のデバイス名を保持できるようになりました。

JPI\$_TERMINAL 項目コードを指定して \$GETJPI を呼び出してこのフィールドをフェッチする場合、特に影響はありません。ただし、システム・サービスに渡すバッファを、最大 16 バイト保持できるように拡張しても構いません。

5.3.7 スレッド単位のセキュリティは特権付きコードとデバイス・ドライバに影響する

V7.3-1

セキュリティ・プロファイルを I/O Request packet (IRP) に添付するために使用する方法が、Version 7.2 で変更されました。

Version 7.2 より前のバージョンの OpenVMS では、IRP 構造体には、要求者のプロセス単位の Access Rights Block (ARB) セキュリティ構造体のアドレスが含まれていました。OpenVMS Alpha Version 7.2 以降、新しいセキュリティ・プロファイル構造体 (Persona Security Block (PSB)) のアドレスが、ARB アドレスの機能置換として、IRP に追加されました。

I/O サブシステムは PSB へのアクセスを、PSB 内のリファレンス・カウンタを通して管理します。I/O サブシステムは、このリファレンス・カウンタを、IRP の作成時にカウントアップし、IRP の I/O 後処理時にカウントダウンします。このカウンタが 0 になったとき、PSB 構造体は割り当て解除されます。

1 つの要求に対して複数の I/O 操作を行うために IRP のコピーを作成またはクローン化して、コピーした IRP を後処理のために I/O サブシステムに渡すデバイス・ドライバは、コードを変更して、追加した IRP 内の PSB への余分な参照に対処しなければなりません。この処理は、コピーされた IRP 内の PSB アドレスを、NSA_STD\$REFERENCE_PSB ルーチンに渡すことで行います。インクルード・ファイルと、NSA_STD\$REFERENCE_PSB の呼び出しは、次のとおりです。

プログラミングに関する注意事項

5.3 特権データ構造体の変更

```
#include <security-macros.h>
/* Increment REFCNT of PSB that is now shared by both IRPs */
nsa_std$reference_psb( irp->irp$ar_psb );
```

デバイス・ドライバは、次の状況でこの変更を行わなければなりません。

- デバイス・ドライバが次の状態の場合
 1. デバイス・ドライバが、既存の IRP を複製することで新しい IRP を作成する場合、かつ
 2. IOC_STD\$SIMREQCOM または IOC_STD\$DIRPOST1 を呼び出すことで、I/O 後処理のためにオリジナル IRP と複製 IRP の両方をキューに登録する場合
デバイス・ドライバは IRP を複製した後、I/O 後処理のためにキューに登録する前に、NSA_STD\$REFERENCE_PSB を呼び出さなければなりません。
- デバイス・ドライバが次の状態の場合
 1. デバイス・ドライバが、既存の IRP を複製することで新しい IRP を作成する場合、かつ
 2. コピーまたはオリジナル IRP の IRP\$L_PID セルにプロシージャ記述子のアドレスを格納しない場合、かつ
 3. IOC_STD\$REQCOM, COM_STD\$POST, COM_STD\$POST_NOCNT, IOC_STD\$POST_IRP を呼び出すことで、I/O 後処理のためにオリジナル IRP と複製 IRP の両方をキューに登録する場合

デバイス・ドライバは IRP を複製した後、I/O 後処理のためにキューに登録する前に、NSA_STD\$REFERENCE_PSB を呼び出さなければなりません。

これらのステップを実行するデバイス・ドライバは、たいていはプロシージャ記述子のアドレスを IRP\$L_PID に格納しています。したがって、IRP を複製するほとんどのデバイス・ドライバは、ソース・コードの変更、再リンク、再コンパイルを行わなくても、OpenVMS Version 7.2 以降で正しく機能するはずですが、

この状態で NSA_STD\$REFERENCE_PSB を呼び出さないと、PSB 内のトラッキング情報が壊れ、システム障害となることがあります。

NSA_STD\$REFERENCE_PSB を呼び出すようにデバイス・ドライバのコードを変更する場合は、OpenVMS Version 7.2 またはそれ以降で動作するように、ドライバを再コンパイルおよび再リンクしなければなりません。

5.3.8 OpenVMS フォーク・スレッド作成のための IPL 要件の強制

V7.3-1

OpenVMS フォーク実行スレッドを作成するためには、いくつかのルーチンを特権コードで使用します。これらのルーチンは、プロセスとは独立にシステム・コンテキストで実行されます。これらのルーチンには4つの形式があり、どの形式を使用するかは、直系のフォークとキューに入れられるフォークのどちらが必要かと、使用している言語インタフェースで決まります。

- EXE\$QUEUE_FORK
- EXE_STD\$QUEUE_FORK
- EXE\$PRIMITIVE_FORK
- EXE_STD\$PRIMITIVE_FORK

これらのルーチンは、実行中に、誤って別のCPUに再スケジュールされないようにするため、IPL\$_RESCHED 以上で呼び出す必要があります。このような再スケジュールが行われると、システムがハングする可能性があります。

OpenVMS V7.3-1 では、SYSTEM_CHECK の値を1にすると、これらのルーチンによって、まずシステムのIPLがチェックされます。IPLがIPL\$_RESCHEDの値よりも小さい場合、システムはSPLINVIPL バグ・チェックで失敗します。

性能上の理由から、SYSTEM_CHECK の値を0にすると(デフォルト)、IPLは検証されません。不正なコードを使用すると、プロセス・コンテキストでこれらのルーチンの実行中に別のCPUへの再スケジュールが発生したときに(IPL\$_RESCHED よりも小さい値を指定した場合など)、システムがハングする可能性があります。

5.4 浮動小数点型データを使用するアプリケーション

V8.2

Itanium®アーキテクチャでは、IEEE 単精度および IEEE 倍精度を含む、IEEE 浮動小数点形式を用いた、ハードウェアによる浮動小数点演算を実装しています。

Alpha ハードウェアでは、IEEE 浮動小数点形式と VAX 浮動小数点形式のいずれもサポートしています。OpenVMS Alpha では、コンパイラはデフォルトで VAX 形式のコードを生成し、オプションで IEEE 形式のコードを生成します。

OpenVMS I64 では、コンパイラはデフォルトで IEEE 形式のコードを生成し、オプションで VAX 形式のコードを生成します。Integrity サーバでは、VAX や Alpha システムで生成された VAX 形式の浮動小数バイナリ・データをアプリケーションで取り扱う必要がある場合を除き、IEEE 形式を使用することをお勧めします。OpenVMS I64 で VAX 形式を使用する場合の詳細については、次の Web サイトのホワイト・ペーパー『Intel® Itanium®における OpenVMS 浮動小数点演算について』を参照してください。

<http://h50146.www5.hp.com/products/software/oe/openvms/technical/>

OpenVMS Version 8.2 では、IEEE 浮動小数点型データを扱うアプリケーション用に、新たに IEEE 浮動小数点型対応版の LIB\$ と OTS\$ のランタイム・ライブラリ・ルーチンを用意しました。システム提供のヘッダ・ファイルには、これらの新しい関数の関数プロトタイプ定義が含まれていません。これらの関数を使用するには、アプリケーションで関数プロトタイプを定義する必要があります。これらの関数の説明は、LIB\$ と OTS\$ のランタイム・ライブラリ・リファレンス・マニュアルを参照してください。これらの関数プロトタイプ定義は、今後リリースされるシステム提供のヘッダ・ファイルに含まれる予定です。

5.5 Ada コンパイラはまだ利用できない (I64 のみ)

V8.2

Ada コンパイラは、OpenVMS Alpha Version 8.2 でサポートされます。弊社では、HP Ada 83 コンパイラを Alpha から I64 に移植する作業は行っていませんが、AdaCore 社で Ada 95 コンパイラを OpenVMS I64 にポーティング中です。この製品が利用可能になる時期については、直接 AdaCore 社にお問い合わせください。

5.6 Backup API: ジャーナリング・コールバック・イベントの制限事項

永続的な制限事項

アプリケーションがジャーナリング・イベントのいずれかに対してコールバック・ルーチンを登録する場合は、すべてのジャーナリング・コールバック・イベントに対してコールバック・ルーチンを登録しなければなりません。ジャーナリング・コールバック・イベントは次のとおりです。

```
BCK_EVENT_K_JOURNAL_OPEN
BCK_EVENT_K_JOURNAL_WRITE
BCK_EVENT_K_JOURNAL_CLOSE
```

コールバック・ルーチンの登録の詳細については、『OpenVMS Utility Routines Manual』の Backup API に関する章を参照してください。

5.7 C プログラム: CASE_LOOKUP=SENSITIVE を設定したコンパイラ

永続的な制限事項

特性として CASE_LOOKUP=CASE=SENSITIVE が設定されているプロセスで C プログラムをコンパイルすると、.h ファイル・タイプ (小文字の「h」) で指定された C プログラム内の #include ファイルは、検出および実行されません。また、システム

の#include ファイルが他の.h ファイル・タイプの#include ファイルを使用している場合、この#include ファイルは検出されず、エラーが出力されます。

この動作を防ぐには、大文字と小文字を区別しないように設定します。case=sensitiveを設定する必要がある場合は、C プログラム内の#include ファイルにファイル・タイプを指定しないか(例#include <stdio>), または大文字のH ファイル・タイプを指定してください(例#include <stdio.H>).

ただし、stdlib.h などのシステム#include ファイルが.h ファイル・タイプの#include ファイルを使用している場合は、エラーとなるので注意してください。

5.8 C ランタイム・ライブラリ

ここでは、C ランタイム・ライブラリ (RTL) の変更や修正について説明します。

5.8.1 socket_fd を使用するプログラムでのメモリ・リークの修正

V8.2

socket_fd を使用するプログラムでメモリ・リークが発生し、ページ・ファイル・クォータを消費してしまう場合があります。この問題は解決されました。

5.8.2 vsnprintf と snprintf によるユーザ・バッファの上書きの修正

V8.2

vsnprintf関数とsnprintf関数は、ユーザ・バッファ内で許されている最大数を超えて、メモリを上書きすることがありました。

たとえば、次のsnprintfの呼び出しは、指定された書式付き文字列を使用し、ユーザ・バッファtで許されているt[2]を超えて、上書きを行っていました。

```
snprintf(t, 3, "%2sxxxx", "a");
```

この問題は解決されました。

5.8.3 mmap と mprotect の変更

V8.2

以前は、OpenVMS Versions 7.3-1 および 7.3-2 用の C RTL を変更すると、mprotect関数用のメモリがすでにmmapでマッピング済みでないと、エラーが返されていました。

この問題は、従来の動作に戻したことで解決されました。再び、`mmap`でマッピングされていないメモリに対して、`mprotect`でプロテクションを設定できるようになりました。

5.8.4 `getpwnam_r` と `getpwuid_r` のポインタの問題の修正

V8.2

`getpwuid_r`のショート・ポインタ版(`_getpwuid_r32`)を呼び出すプログラムが、第3引数の(`buffer`)に対して、誤ってロング・ポインタ値を渡すことがありました。

このユーザ指定のバッファは、結果として出力される`passwd`構造体のメンバ(32ビット・ポインタ)に割り当てられていました。このことにより、バッファがハイ・メモリにある場合は、これらのメンバ(`pw_name`, `pw_dir`, および`pw_shell`)が、誤った結果になっていました。

この問題は、`getpwnam_r` (`_getpwnam_r32`)のショート・ポインタ版でも発生します。

この問題は修正されました。`_getpwnam_r32`と`_getpwuid_r32`のプロトタイプが変更され、`buffer`引数に対して64ビット・ポインタを認めず、32ビット・ポインタだけを受け付けるようになりました。

5.8.5 `_strtok_r32` と `_strtok_r64` がスコープ内になった

V8.2

`<string.h>`をインクルードし、`_strtok_r32`または`_strtok_r64`を呼び出していたプログラムは、スコープ内でプロトタイプを見つけることができませんでした。この問題は修正されました。

5.8.6 `iconv` プロトタイプの、`const` 型修飾子の追加 (Alpha のみ)

V8.2

`XOPEN_SOURCE` 機能テスト・マクロに500が定義されている場合(`#DEFINE XOPEN_SOURCE 500`)、`X/Open` 標準に準拠するために、`<iconv.h>`内の`iconv`関数プロトタイプの第2引数に、`const`修飾子が追加されます。

```
size_t iconv (iconv_t cd, const char **inbuf, size_t *inbytesleft,
              char **outbuf, size_t *outbytesleft);
```

5.8.7 <stddef.h>ヘッダの修正 (Alpha のみ)

V8.2

C++ コンパイラはoffsetofマクロの結果を、整数定数式としては認識していなかったため、このような使い方をすると、C++ コンパイラでエラーとなっていました。

この問題は修正されました。<stddef.h>ヘッダ・ファイルが変更され、offsetofマクロの代わりに定義が、C++ Version 6.5 およびそれ以降で使用するために用意されました。

この代替の定義は、EDG 拡張(__INTADDR__)を使用して、この拡張がなければ非標準の変換である、ポインタから整数への変換を実行します。この解決方法は、EDG が用意して推奨していたもので、害のない名前を使用すること以外は、同じです。

5.8.8 getc マクロの引数が括弧で保護されるようになった (Alpha のみ)

V8.2

ISO C 標準 (ISO/IEC 9899) に準拠するように、getcマクロの引数が、括弧で保護されるようになりました。

5.8.9 インライン関数 getc および getchar の CXXL 接頭辞の問題の修正 (Alpha のみ)

V8.2

C++ でgetcとgetcharをコンパイルすると、リンク時に未定義シンボルの警告が出ていました。特定の状況で、getcマクロ (C の場合) やインライン関数 (C++ の場合) は、実際に関数decc\$getcを呼び出していました。インライン関数getcが、実際に関数decc\$getcの宣言を、extern_prefix "CXXL\$"下で行っていたことが問題でした。同様の問題が、getcharでも発生します。

この問題は修正されました。<stdio.h>ヘッダ・ファイルが修正され、インライン実装の定義が CXXL\$接頭辞の範囲外に残されたまま、getcとgetcharのプロトタイプが、#pragma__extern_prefix "CXXL\$"ディレクティブの範囲内に用意されました。

5.8.10 std 名前空間での非 std 関数の宣言 (Alpha のみ)

V8.2

プログラミングに関する注意事項

5.8 C ランタイム・ライブラリ

`v*scanf`や`*snprintf`関数を参照するソース・ファイルをC++でコンパイルすると、`%CXX-E-UNDECLARED` エラーになっていました。これは、`<stdio.h>`ヘッダ・ファイルがこれらの関数を`std`名前空間で宣言し、グローバル名前空間には入れていなかったためです。

この問題は修正されました。これらの関数宣言は、`std`名前空間外の位置に移動されました。

5.8.11 大きなファイル・オフセットでの `lseek` の問題の修正 (Alpha のみ)

V8.2

`lseek`関数は、`ULONG_MAX` (4294967295) バイトよりも大きいオフセットのファイル位置を正しく指すことができませんでした (`_LARGEFILE` 機能制御マクロ下)。たとえば、`lseek`を6 GB (16 メガブロック)のファイルで呼び出し、オフセットを`0x100000000`と指定すると、ファイル・ポジションは0になっていました。この問題は修正されました。

5.8.12 `<errno.h>`内の新しい `EABANDONED` コード

V8.2

`<errno.h>`ヘッダ・ファイルに、新しい`errno`コードとして`EABANDONED`が追加されました。

`Pthreads` 関数は、ターゲットのプロセス共有タイプの`mutex`が、すでに終了しているプロセスによってロックされている (つまり、権利を持つ所有者が`mutex`を開放できない状態であるため`mutex`が「放棄」された)とシステムで判断した場合、`EABANDONED` (「所有者がリソースを解放できない」)コードを返すことができるようになりました。

5.8.13 `mktime` の問題の修正

V8.2

UTC ベースの関数`mktime`は、構造体のメンバ`tm_mday`の値が0または負のときに日付を算出できませんでした。また、`mktime`は、矛盾した日付を生成することがありました。この問題が修正されています。

5.8.14 <poll.h>内の POLLWRNORM が POLLOUT と等価になった

V8.2

<poll.h>ヘッダ・ファイルに関する X/Open のドキュメントによれば、POLLWRNORM は POLLOUT と等価でなくてはなりません。これまでは、等価ではありませんでしたが、等価になりました。

5.8.15 <in6.h>内の IPV6 構造体がパック形式になった

V8.2

<in6.h>ヘッダ・ファイル内の IPV6 構造体 `sockaddr_in6` は、パック形式ではなかったため、パック形式であると想定してアプリケーションがサイズのチェックをすると問題が発生していました。メンバが自然な境界にアラインされているため、この構造体はパック形式である必要がありました。この問題は修正されました。

5.8.16 <builtins.h>内の __PAL_BUGCHK の問題の修正

V8.2

C または C++ コンパイラを使用するときに、`__PAL_BUGCHK` をパラメータ付きで呼び出すと致命的なエラーになっていました。また、組み込み関数に付いている実装についてのコメントの多くが、誤解を招きやすいものでした。この点が修正されました。

5.8.17 C++ コンパイラの statvfs に関するエラーの修正

V8.2

C, C++ コンパイラ, または X/Open 標準 (XPG6) で型修飾子 `restrict` をサポートしている場合、<decc\$types.h> および <statvfs.h> ヘッダ・ファイルにこの型修飾子が追加されます。

これにより、次の問題が解決されます。

```
int statvfs(const char * __restrict path, struct statvfs * __restrictbuf);
.....^
%CX-E-EXPRPAREN, expected a ")"
```

5.8.18 glob と globfree の問題の修正

V8.2

以下のglobおよびglobfreeに関する問題が修正されました。

- globおよびglobfree関数は、64ビットのポインタをサポートしていませんでした。64ビットのポインタをC RTLに渡すと、(ACCVIOのような)予想のできないう結果になることがありました。これは、C RTLが、64ビットのポインタを受け取することを想定しておらず、globのデータ構造体が32ビット・ポインタのみをサポートしていたからです。
- <glob.h>ヘッダ・ファイルでは、アラインメント・プラグマを指定していませんでした。ユーザが<glob.h>をインクルードしている近くでmember_alignmentを設定すると、glob_t型データのアラインメントが不適切になり、C RTLが予測できない動作をする可能性がありました。

5.8.19 DECC\$SHR_EV56 が正しくリンクされた

V8.2

OpenVMS Version 7.3-2 では、DECC\$SHR_EV56.EXE イメージが正しく最適化されていませんでした。

C RTL は2セットのオブジェクトからビルドされていました。1つは通常どおりにコンパイルされ、もう1つはAlpha EV56 プロセッサ用に最適化されていました。DECC\$SHR_EV56.EXE イメージは、最適化されたオブジェクトでリンクされていませんでした。この問題は修正されました。

5.8.20 Zone Information Compiler (zic) のアップデート

V8.2

新しい時刻インジケータが、Rule 行のAT フィールド用に追加されました。

英字 "u" (または "g", "z") は、AT フィールド内の時刻がUTCであることを示します。

zicについての詳細は、『HP C ランタイム・ライブラリ・リファレンス・マニュアル』を参照してください。

5.9 呼び出し標準規則とローテートするレジスタ (I64 のみ)

V8.2

ここでの説明は『OpenVMS Calling Standard』の情報を補足するものです。

呼び出し標準規則の ICB (invocation context block) (『OpenVMS Calling Standard』の表 4-16 を参照) およびメカニズム・ベクタ (『OpenVMS Calling Standard』の図 8-7 と表 8-6 を参照) は常に、あたかも、レジスタ・リネーム・ベース (CFM.rrb) とローテート・サイズ (CFM.sor) がいずれも 0 であったかのように、汎用レジスタ、浮動小数点レジスタ、およびプレディケート・レジスタを記録しています。言い換えると、ローテートするレジスタを使用しているときに、ローテーションの効果が無視されます。このことは、LIB\$I64_PUT_INVO_REGISTERS ルーチン (『OpenVMS Calling Standard』の第 4.8.3.13 項を参照) で使用するレジスタ・マスクについても同様です。というのは、これらのマスクは、ICB 構造体のフィールドによって定義されるからです。

現在は、補足的なアクセス・ルーチン LIB\$I64_GET_FR, LIB\$I64_SET_FR, LIB\$I64_GET_GR および LIB\$I64_SET_GR (『OpenVMS Calling Standard』の第 4.8.4 項を参照) が、レジスタ・リネーム・ベース・レジスタとローテート・サイズ・レジスタの効果を調整しないで、不適切に、そのレジスタ番号パラメータを解釈しています。これは、誤りであり今後のリリースで修正される予定です。

それまでは、ICB またはメカニズム・ベクタ内の汎用レジスタ、浮動小数点レジスタ、およびプレディケート・レジスタを調べるプログラムや、実行時に見えるレジスタを探して内容を解釈するプログラムでは、保存された CFM レジスタを調べて、自身で適切に調整する必要があります。

5.10 Common Data Security Architecture (CDSA) に関する考慮

ここでは、CDSA に関する注意事項について説明します。

5.10.1 Secure Delivery Advanced Developer's Kit

V8.2

インターネット経由でファイルをダウンロードすることは、手軽にソフトウェアのアップデートをしたいと思う OpenVMS ユーザにとっては、必須条件になりつつあります。一方で、これらのユーザは、セキュリティの脆弱性に用心深くなっています。Secure Delivery は、公開鍵と電子署名技術を用いており、OpenVMS ユーザは HP および他社の OpenVMS ベンダーからダウンロードするファイルが本物であることを確認することができます。

Secure Delivery のについて説明したドキュメントは、次の Web サイトにあります。

<http://www.hp.com/go/openvms/security/>

このドキュメントは、OpenVMS 上の Secure Delivery の概要と、そのコンポーネントの呼び出し方法について説明しています。Secure Delivery は、OpenVMS Version 8.2 の Advanced Developer's Kit (ADK) として提供され、その後の OS リリースで全面的にサポートされる予定です。将来予定されているソフトウェア・キットの自動検証のための PCSI との連携機能を除き、Secure Delivery の機能はすべて OpenVMS Version 8.2 に実装されています。

5.10.2 インストールと初期化に関する考慮

V7.3-2

CDSA は、オペレーティング・システムのインストール時に自動的にインストールされます。ただし、次の点に注意してください。

- CDSA は、OpenVMS Alpha Version 7.3-2 以降のインストール中に同時にインストールされますが、セットアップと初期化は個別に実行する必要があります。CDSA を使用するには、あらかじめ次のプロシージャを手動で実行してください。このプロシージャを実行するには、SYSPRV 特権が必要です。次のコマンドを入力します。

```
$ @SYS$STARTUP:CDSA$INITIALIZE
```

新しいバージョンの CDSA をインストールする場合は(たとえば、フィールド・テストから運用バージョンへ、または新しいバージョンの OpenVMS へのアップグレードにおいて)、CDSA のアップグレード・プロシージャ (@SYS\$STARTUP:CDSA\$UPGRADE) を実行する必要があります。アップグレード・プロシージャを実行する前に、すべての CDSA アプリケーションをシャットダウンしてください。

システムをリブートするときに、初期化プロシージャやアップグレード・プロシージャを再実行する必要はありません。また、OpenVMS スタートアップ・プロシージャに初期化プロシージャやアップグレード・プロシージャを追加する必要はありません。

- 使用中のシステムから CDSA を削除しないでください。CDSA を削除するように見えるオプションがありますが、CDSA に対する PCSI コマンドの PRODUCT REMOVE はサポートされていません。(このオプションはインストールに使用する PCSI ユーティリティの機能です)。CDSA とオペレーティング・システムは、同時にインストールされ、密接に結合しているため、OpenVMS から CDSA を削除しようとする、正常に動作しないで、望ましくない影響が発生する可能性があります。CDSA を削除しようとする、次のようなメッセージが表示されません。

```
%PCSI-E-HRDREF, product CPQ AXPVMS CDSA Vn.n is referenced  
by DEC AXPVMS OPENVMS V8.2  
-PCSI-E-HRDRF1, the two products are tightly bound by this  
software dependency
```

5.11 照合キーを指定する CONVERT/NOSORT で発生する可能のある CONVERT-I-SEQ エラー

V7.3

/NOSORT 修飾子および出力ファイルのいずれかのキーに照合キー・タイプを指定する CONVERT コマンドでは、潜在的な動作の変更による影響があります。

CONVERT コマンドの/NOSORT 修飾子は、主キーが入力ファイルですすでにソート順に並べられているため、Convert ユーティリティでソートしないことを表します。OpenVMS Version 7.3 より前のバージョンの Convert ユーティリティでは、出力ファイルに指定されたキーに照合キー・タイプが含まれる場合、/NOSORT が指定されているかどうかに関係なく、入力ファイルが必ずソートされるという欠陥がありました。OpenVMS Version 7.3 では、Convert ユーティリティが修正され、出力ファイルのいずれかのキーが照合キーである場合でも、コマンド行の/NOSORT 修飾子に正しく従うように修正されています。

そのため、以前に照合キー欠陥の 2 次的な影響で成功していた CONVERT 操作でも、入力ファイルが主キーによってソートされていない場合で、コマンド行に/NOSORT が指定されているときには、%CONVERT-I-SEQ メッセージが表示されることがあります。入力ファイルが主キーによってソートされていない場合は、/NOSORT 修飾子は、使用しないでください。

5.12 デバッグ・モード: CPUSPINWAIT バグ・チェックの回避

V7.3-1

OpenVMS オペレーティング・システムには、複雑なハードウェアの問題やソフトウェアの問題をデバッグするのに役立つように、多くの特殊操作モードが準備されています。一般には、これらの特殊モードを使用すれば、特別なレベルでトレース、データの記録、一貫性チェックを行うことができ、このような機能は、問題があるハードウェア構成要素やソフトウェア構成要素を突き止めるのに役立ちます。これらの操作モードは、システム・パラメータ MULTIPROCESSING, POOLCHECK, BUGCHECKFATAL, SYSTEM_CHECK によって制御されます。

一般に I/O 負荷の高い特定の状況で、これらの特殊モードのいずれかを使用している場合は(たとえば、デバイス・ドライバや他の複雑なアプリケーションをデバッグする場合など)、CPUSPINWAIT バグ・チェックが発生することがあります。特に、スピンロックのある状態で長期間実行する特権コードに対して CPUSPINWAIT バグ・

チェックが発生します。スピンロックは、クリティカル・セクションのエントリ・ポイントとイグジット・ポイントを区切るために使われ、この場合のように連続的に使うことはできません。

CPUSPINWAIT バグ・チェックを防止するには、これらのシステム・パラメータに対して、システムのデフォルト設定を使用するか、またはシステムの負荷を低下させます。

何らかの理由でデフォルトの設定を変更しなければならない場合は、SMP_LNGSPINWAIT システム・パラメータを 9000000 に設定することで、問題が発生する可能性を減らせます。

5.13 Delta/XDelta デバッグ

ここでは、OpenVMS Alpha および I64 システム上で動作する OpenVMS Delta および XDelta デバッグに関する注意事項について説明します。

OpenVMS Debugger に関する注意事項は第 5.28 節を参照してください。

5.13.1 Delta デバッグは OpenVMS I64 では利用できない

V8.2

Delta デバッグは、OpenVMS I64 オペレーティング・システムにはまだ移植されていません。

5.13.2 I64 システムでの XDelta の制限事項

V8.2

次の Intel® Itanium®ハードウェア・レジスタは、XDelta ではサポートされていません。

- CPUID
- Debug Data Break レジスタ
- Debug Instruction Break レジスタ
- Region レジスタ
- Protection Key レジスタ
- Instruction Translation レジスタ
- Data Translation レジスタ
- Device Interrupt Control レジスタ

5.13.3 I64 システムと Alpha システムでの XDelta の相違点

V8.2

OpenVMS I64 上で実行中のシステムに割り込むには、システム・コンソールで Ctrl/P を押します。XDelta は、あらかじめロードされていなければなりません。Ctrl/P を押すと、現在の PC と現在の IPL でシステムが停止します。OpenVMS Alpha システムと異なり、IPL が 14 未満になるのを待つための遅延はありません。

5.13.4 XDelta のレジスタ表示に関する考慮 (I64 のみ)

V8.2

OpenVMS I64 上の XDelta は、あたかも、レジスタ・リネーム・ベース (CFM.rrb) とローテート・サイズ (CFM.sor) がいずれも 0 であるかのように、汎用レジスタ、浮動小数点レジスタ、およびプレディケート・レジスタを表示します。言い換えると、ローテートするレジスタを使用しているときには、ローテーションの効果は無視されます。この状況は、今後のリリースで修正される予定です。詳細は第 5.9 節を参照してください。

5.14 ファイル・アプリケーション: 『Guide to OpenVMS File Applications』の訂正

V8.2

マニュアル『Guide to OpenVMS File Applications』が改訂されるときには、下記の訂正が盛り込まれます。

- 表 1-4 は誤解を招く可能性があります。ファイル・フォーマットの比較表で、ODS-2 や ODS-5 のファイル・フォーマットでのディレクトリの制限として 256 としてあります。この説明は、256 ディレクトリ・レベルとし、ユーザが、ディレクトリ数が 256 に制限されているものと誤解しないようにする必要があります。

『OpenVMS システム管理者マニュアル』の類似した表は Version 8.2 で訂正されました。

- 第 6.6.3 項の第 4 パラグラフ (Note はカウントしない) を、以下の内容に置き換えます。

「プログラム内で使用するルート・デバイス論理名を、SET DEFAULT コマンドで定義する際には、DCL コマンドの DEFINE または ASSIGN で /TRANSLATION_ATTRIBUTES=CONCEALED 修飾子を使用して、この論理名が隠し装置の論理名であることを指定します。隠し装置の論理名をルート・デバイス論理名として定義するためには、ルート・ディレクトリがピリオド (.) で終わっていなければなりません (例: DUA22:[ROOT.])。また、デバイスの指定が、

物理デバイス名でなければなりません。ルート・デバイス論理名の等価名には、他の論理名を含めることはできません。ディレクトリを指定するときには、ルート・ディレクトリに対して、後のピリオドだけを使用することができます。」

5.15 RMS 構造体についての HP BLISS コンパイラの警告 (I64 のみ)

V8.2

RMS ユーザ構造体 (たとえば, FAB, RAB) の割り当てに使用できる BLISS マクロ (\$xxx_DECL) に、キーワード・アラインメントが追加されました。プロセッサが高速になるほど、アラインメント・フォルトは性能に悪影響を及ぼします。アラインメントをマクロ内に直接実装することにより、これらのマクロを使用する、BLISS で書かれた多数の OpenVMS ユーティリティおよびユーザ・アプリケーションは、性能が改善されます。

該当するマクロは、\$FAB_DECL, \$NAM_DECL, \$NAML_DECL, \$RAB_DECL, \$RAB64_DECL, \$XABALL_DECL, \$XABDAT_DECL, \$XABFHC_DECL, \$XABITM_DECL, \$XABJNL_DECL, \$XABKEY_DECL, \$XABPRO_DECL, \$XABRDT_DECL, \$XABRU_DECL, \$XABTRM_DECL, および \$XABSUM_DECL です。

RMS マクロに追加されたアラインメントにより、コンパイラがアラインメント競合の警告を出力することがあります。コンパイラの警告があるプログラムでも、正しくリンクして、実行することができます。ただし、ソースに簡単な変更を加えて、警告を取り除くことをお勧めします。

これらのマクロを BLISS アプリケーション内で使用し、宣言に ALIGN 属性が含まれている場合、BLISS コンパイラは“conflicting or multiply specified attribute”という警告を出力します。たとえば、FAB: \$FAB_DECL ALIGN(2) という宣言に対して、警告が出力されます。この警告は、キーワード・アラインメント (ALIGN(3)) を指定したとしても出力されます。これらのマクロに関連する明示的な ALIGN 属性を削除する必要があります。

さらに、これらの割り当てが、ALIGN(3) と競合する明示的なアラインメント (ALIGN(3) 未満のもの) を持つ PSECT に含まれている場合、BLISS コンパイラは、“align request negative or exceeds that of psect”という警告を出力します。たとえば、次の宣言に対して警告が出力されます。

```
PSECT OWN = $OWN$ (... , ALIGN(2), ...)
OWN
    FAB = $FAB_DECL, ...
```


BLISS アプリケーションの再コンパイル時に PSECT のアラインメントに関する警告が表示された場合、PSECT のアラインメントを ALIGN(3) (またはそれ以上) に調整してください。まれに、PSECT 間でデータが隣接していると仮定しているアプリケーションが存在することがあります。この変更により、このような仮定が成り立たなくなる場合があります。そのため、コードでこのような仮定を行っていないかチェックし、必要な変更を行ってください。

多くの OpenVMS ユーティリティが BLISS で記述されていますが、OpenVMS 全体のビルドで発生した警告はわずかでした。この警告を取り除くために OpenVMS に加えた変更は他にはありませんでした。このため、修正が必要なユーザ・アプリケーションは、ほとんどないと考えられます。

5.16 HP COBOL ランタイム・ライブラリ (RTL)

V8.2

OpenVMS Alpha および OpenVMS I64 Version 8.2 では、HP COBOL RTL (DEC\$COBRTL) は V2.8-775 にアップデートされました。

5.16.1 SET および COB\$SWITCHES

V8.2

COBOL の SET 文の実装 (論理名 COB\$SWITCHES を使用して実装されている) は、LNM\$FILE_DEV 内の最初のエントリを使用するのに失敗したときに、LNM\$PROCESS を使用するように変更されました。これまでは、LNM\$FILE_DEV 内の最初のエントリが書き込み不能な論理名テーブルであると、COBOL の SET 文はフェールしていました。

5.16.2 レコード・ロックの問題の修正

V8.2

COBOL の START 文または WRITE 文で、自動レコード・ロックで複数のレコード・ロックが取得されることがありました。この問題は修正されました。

5.17 HP Decimal Support ランタイム・ライブラリ (RTL)

V8.2

OpenVMS Alpha および OpenVMS I64 Version 8.2 では、HP Decimal Support RTL (LIBOTS2) が V2.8-67 にアップデートされました。

5.18 I64 用の HP Fortran

V8.2

OpenVMS I64 Fortran コンパイラは、OpenVMS Alpha の HP Fortran 90 を移植したものです。このコンパイラは OpenVMS I64 システム上で動作し、OpenVMS I64 システム用のオブジェクトを生成します。このオブジェクトは、OpenVMS I64 上の標準リンカを使用してリンクされます。このコンパイラは、OpenVMS I64 Version 8.2 を必要とします。

OpenVMS I64 システム用の HP Fortran のコマンド行オプションと言語機能は、以下の例外を除いて、OpenVMS Alpha システム用の HP Fortran 90 のものと同じです。

- 浮動小数点の計算

要点は、ホワイト・ペーパー『Intel® Itanium®アーキテクチャにおける OpenVMS 浮動小数点演算について』を参照してください。このドキュメントは、次の Web サイトで参照できます。

<http://h71000.www7.hp.com/openvms/integrity/resources.html>

- デフォルトの浮動小数点データ型は IEEE です。(つまり、/FLOAT=IEEE_FLOAT がデフォルトです)。
- /IEEE_MODE 修飾子のデフォルト値は、/IEEE_MODE=DENORM_RESULTS です。
- ユーザは/FLOAD の値を 1 つと、/IEEE_MODE の値を 1 つ選択し、アプリケーション全体でその値を維持しなければなりません。
- F90 コンパイラだけがサポートされています。以前/OLD_F77 修飾子で起動されていた F77 コンパイラは利用できません。FDML や CDD のサポートなど、Alpha F77 コンパイラに含まれていて、Alpha F90 コンパイラでは利用できない一部の機能が、I64 F90 コンパイラに実装されています。詳細は、Fortran T8.0 製品のリリース・ノートを参照してください。
- /ARCH 修飾子と/TUNE 修飾子での値 Alpha は、コンパイル・アンド・ゴアの互換性を保つために、コンパイラ起動コマンドで利用することができます。これらの値を無視したことを示す情報メッセージが表示されます。

インストール手順など、このリリースについての詳細は、Fortran T8.0 製品のリリース・ノートを参照してください。リリース・ノートを抽出するには、Fortran PCSI キットが置かれているディレクトリをデフォルトとして設定し、次のコマンドのいずれかを入力します。

```
$ PRODUCT EXTRACT RELEASE_NOTES FORTRAN ! For TXT file  
$ PRODUCT EXTRACT FILE FORTRAN/SELECT=FORTRAN_RELEASE_NOTES.PS ! For PS file
```

5.19 OpenVMS 用 HP MACRO

OpenVMS MACRO コンパイラは、OpenVMS VAX システム用に記述された Macro-32 ソース・コード (VAX MACROアセンブラ) をコンパイルし、OpenVMS Alpha および OpenVMS I64 システムで動作する機械語コードに変換します。ここでは、MACRO コンパイラに関する注意事項について説明します。

5.19.1 OpenVMS I64 用 HP MACRO

V8.2

OpenVMS I64 Version 8.2 では、ネイティブの MACRO コンパイラが用意されています。このコンパイラは、OpenVMS Alpha 上の MACRO コンパイラと同じ DCL コマンドを使用します。

OpenVMS Alpha 上でコンパイルした大半のプログラムは、変更せずにそのまま OpenVMS I64 上で再コンパイルできるはずですが、ただし、非標準の戻り値を持つルーチン呼び出すプログラムや、他の言語で記述された JSB 命令呼び出しルーチンを使用するプログラムは、ソース・ファイルにいくつかの新しいディレクティブを追加しなければなりません。

新しいディレクティブの詳細や、MACRO コンパイラの詳細は、『OpenVMS MACRO-32 Porting and User's Guide』を参照してください。

5.19.2 /TIE 修飾子のデフォルトは Alpha と I64 で異なる

V8.2

/TIE 修飾子のデフォルトは Alpha システムと I64 システムで異なります。Alpha では、デフォルトは/TIE です。I64 では、デフォルトは/NOTIE です。

5.19.3 /OPTIMIZE=VAXREGS 修飾子は I64 ではサポートされない

V8.2

Alpha システムでサポートされていた/OPTIMIZE=VAXREGS 修飾子は、I64 システムではサポートされません。残念ながら、関連するコードすべてがコマンド行処理から削除されてはなりません。I64 システムで/OPTIMIZE=ALL を指定すると、サポートされていない VAXREGS 最適化を誤って起動することになります。今後のリリースで、コマンド行プロセスを修正して VAXREGS を起動しないようにする予定です。

5.19.4 CODGENWARN メッセージは無視できる (Alpha のみ)

V8.2

Macro-32 コンパイラが、次のようなメッセージを出力することがあります。

```
%AMAC-W-CODGENWARN, pre-allocation of multiple condition codes overlapped
```

このメッセージは無視して構いません。生成されたコードは正しいものです。このようなメッセージが出力されないよう、Macro-32 コンパイラは今後のリリースで修正される予定です。

5.19.5 操作単位のサポート (I64 のみ)

V8.2

Macro-32 コンパイラは、/GRANULARITY 修飾子や.PRESERVE GRANULARITY オプションを完全にはサポートしていません。多くの場合、Macro-32 コンパイラはデフォルトでバイトを操作単位とするコードを生成しますが、そうでない場合 (たとえば、整列されていないクォードワードにアクセスしている場合) があります。この問題は、今後のリリースで修正される予定です。

5.19.6 デフォルトでは整数のゼロ除算エラーは検出されない (I64 のみ)

V8.2

Macro-32 コンパイラは、整数のゼロ除算エラーを正しく検出するコードを生成しません。/ENABLE=OVERFLOW 修飾子を指定すると、Macro-32 コンパイラは実行時の値での除算に関してゼロ除算をチェックするコードを生成しようとします。しかし、これは間違ったオペランドをチェックします。ゼロ以外の数値でゼロを除算すると、このコードはエラーを発生します。今後のリリースでコンパイラは修正され、デフォルトでゼロ除算を正しく検出するコードを生成するようになります。動作は、現在の OpenVMS VAX および OpenVMS Alpha システムでの動作と同じになります。

5.19.7 整数を負の最大値で除算するとコンパイラがクラッシュする (I64 のみ)

V8.2

次の命令で Macro-32 コンパイラは、クラッシュします。

```
DIVL3 R0, #^X80000000, R0  
BVS 1$
```

この問題は、今後のリリースで修正される予定です。

5.19.8 整数の除算で“V”コンディション・コードが正しくセットされない (I64 のみ)

V8.2

整数の除算命令 (DIVL, DIVW, および DIVB) で、オーバーフロー (“V”) コンディション・コードが正しくセットされないことがあります。この問題は、今後のリリースで修正される予定です。

5.19.9 浮動小数点数のゼロ除算エラーが検出されない (I64 のみ)

V8.2

Macro-32 浮動小数点数サポート・ルーチンは、浮動小数点数のゼロ除算を検出しません。サポート・ルーチンでは、VAX 浮動小数点数を IEEE 浮動小数点数に変換して除算を実行します。チェック処理無しで、除算で IEEE の NaN 値 (非数値) が生成されます。サポート・ルーチンは、次に NaN 値を VAX 浮動小数点数に戻そうとします。この操作で、不正浮動小数点数のエラーになります。今後のリリースで、サポート・ルーチンが修正され、正しく浮動小数点数のゼロ除算エラーを検出するようになります。

5.19.10 INSV 命令でメモリを余計に上書きする (I64 のみ)

V8.2

サイズ引数がリテラル 32 で、ポジション引数が 0 より大きいとき、INSV 命令用に生成されたコードが、必要以上にメモリを上書きします。そのようなケースでは INSV 命令が、2 番目のロングワード (メモリ・デスティネーションの場合)、または 2 番目のレジスタ (レジスタ・デスティネーションの場合) を必要以上に上書きします。

デスティネーションがメモリ参照である場合、リテラル 32 をスクラッチ・レジスタに入れ、そのスクラッチ・レジスタを INSV 命令のサイズ・オペランドとして使用することで、この問題を回避できます。

たとえば、次のようなコードを考えます。

```
INSV R2, #15, #32, (R4)
```

このコードは、R4 が指すロングワードの上位部分を正しくアップデートしますが、次のロングワードの下位部分のみに書き込むのではなく、次のロングワード全体に不正に書き込みます。次のように変更すれば、正しく動作します。

```
MOVL #32, Rtmp ; Rtmp は使用可能なレジスタ  
INSV R2, #15, Rtmp, (R4)
```

デスティネーションがレジスタ参照である場合，回避策は INSV 命令を，複数の命令に分割することです。たとえば次のコードを

```
INSV R2, #15, #32, R4
```

次のように変更します。

```
INSV R2, #15, #17, R4 ; R2 の下位 17 ビットを R4 の上位に  
ASHL #-15, R2, Rtmp ; R2 の上位部分を Rtmp に  
INSV Rtmp, #0, #15, R5 ; R2 の上位 15 ビットを R5 の下位に
```

この問題は，今後のリリースで修正される予定です。

5.20 Hypersort ユーティリティ

ここでは，OpenVMS Alpha および OpenVMS I64 Version 8.2 用の Hypersort V08-006 に関する注意事項について説明します。

Hypersort で修正されていない問題を回避する場合，または Hypersort に実装されていない機能を使用する場合には，従来どおり SORT32 を使用してください。SORT32 に関する注意事項は第 5.36 節を参照してください。

5.20.1 弊社への問題の報告

V8.2

SORT や MERGE で問題を発見した場合は，問題を報告する前に，次のコマンドを実行してください。

```
$ WRITE SYS$OUTPUT "WSEXTENT = 'F$GETJPI(,"WSEXTENT")'"  
$ WRITE SYS$OUTPUT "PGFLQUOTA = 'F$GETJPI(,"PGFLQUOTA")'"  
$ SHOW LOGICAL SORTSHR  
$ SORT/STATISTICS (or MERGE/STATISTICS)
```

問題を再現する入力ファイルとともに，この出力を問題の報告に含めてください。

5.20.2 ラージ・ファイルの制限事項

V8.2

Hypersort V08-010 は，メモリ割り当てのアルゴリズムが改良されましたが，大規模な入力ファイルが使用されると，ハングアップしたり，ACCVIO が発生することがあります。ハングアップや ACCVIO が発生する可能性を低くするには，第 5.20.8 項に記載されているとおりにページ・ファイル・クォータとワーキング・セット・エクステントを設定します。Hypersort でハングアップしたり ACCVIO が発生する場合は，代わりに SORT32 を使用してください。

5.20.3 Hypersort と VFC ファイルの制限事項

V7.3-2

Hypersort で VFC ファイルを使用するには、`/FORMAT=RECORD_SIZE:n` が必要です。

5.20.4 /FORMAT=RECORD_SIZE の制限事項

V7.3-1

Hypersort では、SORT と MERGE の両方で使用する `/FORMAT=RECORD_SIZE:n` がサポートされます。ただし、次の 2 つの制限事項があります。

- すべての場合において、コマンドで指定した RECORD_SIZE の値が入力ファイル内の任意のレコードの最大レコード長 (LRL) よりも小さい場合、長すぎるレコードは、ソートされた出力ファイルで RECORD_SIZE のサイズまで切り捨てられ、診断メッセージ `%SORT-E-BAD_LRL` が発行されます。この場合は、出力ファイルを破棄し、ソートを再実行する必要があります。SORT コマンドの RECORD_SIZE パラメータの値を、DIR/FULL コマンドを実行して表示される入力ファイルの最大レコードのサイズに合わせて修正してください。
- SORT や MERGE によって、入力索引順編成ファイルから出力順編成ファイルが作成されます。この場合、`%SORT-E-BAD_LRL` 診断メッセージも発行される場合があります。

5.20.5 Hypersort と検索リスト、および論理名の使用

V7.3-1

Hypersort では、検索リスト、および入力ファイルと作業ファイルで使用される論理名のサポートが十分ではありません。この問題を検出した場合は、SORT32 を使用してください。

5.20.6 作業ファイルの空き領域不足

V7.3-1

すべてのソート作業ファイルで空き領域が無くなると、Hypersort が正しく終了しません。この問題を防ぐには、次のいずれかの処理を実行してください。

- ソートの作業ファイル用に使用するデバイスに、十分な空き領域を割り当てる。
- SORT32 を使用して、作業ファイルの領域を使い果たしたことを検出する。

5.20.7 入力アスタリスク (*) の制限事項

V7.3

Hypersort では、入力ファイル指定にアスタリスク (*) を使用できません。

5.20.8 最適化されたワーキング・セット・エクステントとページ・ファイル・クォータの設定

V7.3-1

SORT32 と Hypersort は、異なるソート・アルゴリズムと作業ファイル・アルゴリズムを使用します。これらのユーティリティの相対的な速度は、入力ファイルと、メモリ、ディスク、および CPU の構成によって異なります。ワーキング・セット・エクステントが、ページ・ファイル・クォータの 3 分の 1 を超えないようにしてください。SORT32 と Hypersort はいずれも、個々の入力ファイルに合ったワーキング・セット・エクステントを使用することで、最高の性能を発揮します。

5.21 Intel®アセンブラ (I64 のみ)

V8.2

Intel アセンブラ言語で記述されたすべてのモジュールには、-Xunwind フラグによる自動生成を行うか、ソース・モジュールに明示的に unwind ディレクティブを記述する方法で、適切な unwind ディレクティブが含まれていなければなりません。正確な unwind 情報なしでは、オペレーティング・システムの条件処理と例外ディスパッチが動作せず、予期しない状態でプログラムが失敗することがあります。正確な unwind 情報を持たないプログラムは、OpenVMS ではサポートされていません。この前提条件は、永続的な要件となります。

5.22 Librarian ユーティリティ

ここでは、Librarian ユーティリティと Library Service ルーチンに関する注意事項について説明します。

5.22.1 data-reduced ELF オブジェクト・ライブラリとのリンクは推奨できない (I64 のみ)

V8.2

DCX data-reduced ELF オブジェクト・ライブラリは、ライブラリ内の連続領域には格納されません。その結果、最初にデータの展開を行う必要があるため、モジュールをプロセスの P2 空間に直接マッピングすることはできません。LBR\$MAP_

MODULE ライブラリ・サービスが、モジュールを展開しプロセスの P2 空間にコピーします。この動作により、結果としてできた P2 空間内のページが、プロセス・クォータとしてカウントされます。

LBR\$UNMAP_MODULE ライブラリ・サービスは、これらのページを回復しますが、これらのページはヒープ領域解放リストに残り、プロセス・クォータとしてカウントされ続けます。そのため、Linker 操作の前に、あらかじめ DCX data-reduced ELF オブジェクト・ライブラリを展開することをお勧めします。

5.22.2 I64 ライブラリへの.STB ファイルの挿入または置き換えの失敗 (I64 のみ)

V8.2

OpenVMS VAX および Alpha では、.STB ファイルをオブジェクト・ライブラリに格納することができます。OpenVMS I64 では、Librarian はこの動作を行いません。例を次に示します。

```
$ LIBR/CREATE OBJ$:SOME_LIBRARY OBJ$:SOME_FILE.STB
Librarian T01-23
%LIBRAR-E-NOTELFFILE, TPSSWRKD$: [TPSS.TRACE.IA64.LPIOBJ]TRACEMSG.STB;1
  is not an ELF object or image file
```

.STB ファイルはオブジェクトでもイメージでもないため、現在 Librarian は、このファイルを I64 システムの.OLB ファイルには格納しません。

ただし、Alpha と VAX では、.STB ファイルはオブジェクト・ファイルと同様の形式になっています。VAX では、.STB ファイルを Linker への入力として使用することもできます。Alpha では、.STB ファイルの形式は、.OBJ ファイルの形式と同じです（つまり、ファイル拡張子が.STB であること以外は、これらのファイルに違いはありません）。そのため、Alpha の Librarian はこのファイルを受け付けますが、Linker の入力として使用することはできません。

I64 では、.STB ファイルにファイル・タイプ (ET_VMS_LINK_STB) が埋め込まれます。これにより、Librarian と Linker が、.STB ファイルを処理できないことを判断できます。

5.22.3 プロセス・クォータが低すぎると Librarian がエラーを通知しない問題

V8.2

OpenVMS Alpha および I64 の Librarian は圧縮、データ・リダクション、データ拡張操作でエラーを通知しないことがあります。この問題が発生するのは、Librarian が動作しているアカウントまたはプロセスの PGFLQUOTA プロセス・クォータが低い場合です。\$PUTMSG システム・サービスは、エラーが発生した場合でも必ず SS\$_NORMAL というステータスを返すので、操作エラーがただちに明らかになりま

せん。しかし、エラーが発生した場合には、Librarian は Success 以外のステータスを返します。

この問題を回避するには、PGFLQUOTA を増加させてから圧縮、データ・リダクション、データ拡張操作を実行します。さらに、コマンド・プロシージャで LIBRARY コマンドからの戻りステータスを確認するようにしてください。

5.22.4 オブジェクト・モジュール名の長さの問題 (I64 のみ)

V8.2

Librarian は、I64 (ELF) オブジェクトと共有イメージ・ライブラリのモジュール名を誤って 31 文字までに制限しています。この問題は、今後のリリースで修正される予定です。イメージの名前に関する規定については『HP OpenVMS Version 8.2 新機能説明書』を参照してください。

5.23 Lightweight Directory Access Protocol (LDAP) API

ここでは、LDAP API に関する注意事項について説明します。

5.23.1 ld が NULL の場合の ldap_get_option ルーチンからのエラー戻り値 (-1)

V7.3

ldap_get_options() の呼び出しで ld パラメータに NULL 値を使用すると、グローバル・デフォルト・データ・セットではなく、エラー戻り値 -1 が返されます。

5.23.2 ber_flatten() ルーチンが中括弧の不一致を検出しない

V7.3

ber_flatten() ルーチンでは、BerElement 内の '{' および '}' 形式修飾子が一致しない場合が正しく検出されません。

5.24 OpenVMS Alpha 用 Linker ユーティリティ

ここでは、Alpha Linker ユーティリティに関する注意事項について説明します。

I64 Linker に関する注意事項は第 5.25 節を参照してください。

5.24.1 LINK/NATIVE_ONLY のヘルプ・テキストの明確化

V8.2

LINK/NATIVE_ONLY のオンライン・ヘルプを、Alpha および I64 に関してより明確に記述すると、次のように記述することができます。

“リンカがプロシージャ・シグネチャ情報を生成するのを防ぎます。リンクするネイティブ・コードを、VAX または Alpha のバイナリ・コードから変換されたイメージと一緒に使用するためにはプロシージャ・シグネチャが必要です。実行イメージまたは共有イメージが、変換されたイメージを呼び出したり、変換されたイメージから呼び出されたりできるようにするには、/NONATIVE_ONLY を使用してリンクしてください。変換されたイメージと一緒に使用するコードは、/TIE 修飾子を使用してコンパイルしなくてはなりません。(詳細は、関連するコンパイラのマニュアルを参照してください。)”

5.24.2 多数のファイルを指定した場合に Linker がハングアップしたように見える

V7.3-2

RMS_RELATED_CONTEXT リンカ・オプションがオン(デフォルトは RMS_RELATED_CONTEXT=YES)で、存在しないファイルが LINK コマンドのファイル・リストに指定されていた場合、リンカによる LIB\$FIND_FILE の呼び出しは完了するまでに長時間かかり、リンカがハングアップしたように見えます。リンクしているファイルの数と、ファイル指定での論理名の使用状況に応じて、リンカの処理が完了するまでに数時間かかることもあります。これは LIB\$FIND_FILE が、不明ファイルについて接頭辞の組み合わせをすべて探してから、“file not found”メッセージを表示するためです。リンカが LIB\$FIND_FILE を呼び出した後は、Ctrl/Y を押してもリンカを終了させることはできません。

回避方法

LINK コマンドに SYS\$INPUT:/OPTION を指定します。Return を押すと、リンカはユーザがオプション・ファイルに情報を入力するのを待ちます。入力終了したら、Ctrl/Z を押します。この問題を回避するには、オプション・ファイルに次の項目を入れます。

- 1 行目に、次のオプションを指定します。

```
RMS_RELATED_CONTEXT=NO
```

RMS_RELATED_CONTEXT オプションに NO を設定した場合、このオプション・ファイルにリストされている不明ファイルに対しては、すぐに “file not found” メッセージが生成されます。

- 以降の行では、リンクするファイルすべてを、完全ファイル指定 (*disk:[dir]filename.ext* の形式) で指定します。RMS_RELATED_CONTEXT=NO

プログラミングに関する注意事項

5.24 OpenVMS Alpha 用 Linker ユーティリティ

を指定するとファイル名の省略指定ができず、完全ファイル指定が必要になります。

たとえば、次の LINK コマンドがあるとします。

```
$ LINK DSK:[TEST]A.OBJ, B.OBJ
```

このコマンドを RMS_RELATED_CONTEXT=NO とともに指定したい場合は、/OPTION を指定してから、リンクするファイルの完全ファイル指定を、次のように入力します。

```
$ LINK SYS$INPUT:/OPTION
RMS_RELATED_CONTEXT=NO
DSK:[TEST]A.OBJ, DSK:[TEST]B.OBJ Ctrl/Z
$
```

RMS_RELATED_CONTEXT オプションについての詳細は、『OpenVMS Linker Utility Manual』を参照してください。

例

次の例では、リストに DOES_NOT_EXIST.OBJ ファイルが含まれていて、RMS_RELATED_CONTEXT オプションが指定されていない (デフォルトで YES が使用されます) 場合に、リンクがハングアップしたように見える様子を示しています。

```
$ DEFINE DSK$ WORK4:[TEST.LINKER.OBJ.]
$ DEFINE RESD$ ROOT$, ROOT2$, ROOT3$, ROOT4$, ROOT5$, DISK_READ$:[SYS.] 1
$ DEFINE ROOT$ WORK4:[TEST.PUBLIC.TEST]
$ DEFINE ROOT2$ WORK4:[TEST.LINKER.]
$ DEFINE ROOT3$ WORK4:[TEST.UTIL32.]
$ DEFINE ROOT4$ WORK4:[TEST.PUBLIC.]
$ DEFINE ROOT5$ WORK4:[TEST.PUBLIC.TMP]
$ LINK/MAP/FULL/CROSS/EXE=ALPHA.EXE RESD$:[TMPOBJ] A.OBJ,-
_$ RESD$:[SRC]B.OBJ,C,DSKD$:[OBJ]D.OBJ,E,RESD$:[TMPSRC]F.OBJ,-
_$ RESD$:[TEST]G.OBJ,RESD$:[SRC.OBJ]H,RESD$:[COM]DOES_NOT_EXIST.OBJ
Ctrl/T NODE6::_FTA183: 15:49:46 LINK CPU=00:02:30.04 PF=5154 IO=254510 MEM=134 2
Ctrl/T NODE6::_FTA183: 15:49:46 LINK CPU=00:02:30.05 PF=5154 IO=254513 MEM=134
Ctrl/T NODE6::_FTA183: 15:50:02 LINK CPU=00:02:38.27 PF=5154 IO=268246 MEM=134
Ctrl/T NODE6::_FTA183: 15:50:02 LINK CPU=00:02:38.28 PF=5154 IO=268253 MEM=134
Ctrl/T NODE6::_FTA183: 15:50:14 LINK CPU=00:02:44.70 PF=5154 IO=278883 MEM=134
```

- 1 このコマンドは、論理名と、それに対応する文字列を定義しています。
- 2 Ctrl/T を押すたびに、CPU 値と IO 値が大きくなります。ただし、MEM 値と PF 値は大きくなりません。これは、LIB\$FIND_FILE が呼び出されていることを示しています。

次の例のように、オプション・ファイルを使用して RMS_RELATED_CONTEXT に NO を設定すると、不明ファイルを見つけたときにリンク操作がすぐに終了します。

```
$ DEFINE DSKD$ WORK4:[TEST.LINKER.OBJ.]
$ DEFINE RESD$ ROOT$, ROOT2$, ROOT3$, ROOT4$, ROOT5$, DISK_READ$:[SYS.]
$ DEFINE ROOT$ WORK4:[TEST.PUBLIC.TEST.]
$ DEFINE ROOT2$ WORK4:[TEST.LINKER.]
$ DEFINE ROOT3$ WORK4:[TEST.UTIL32.]
$ DEFINE ROOT4$ WORK4:[TEST.PUBLIC.]
$ DEFINE ROOT5$ WORK4:[TEST.PUBLIC.TMP.]
$ LINK/MAP/FULL/ CROSS /EXE=ALPHA.EXE SYS$INPUT:/OPTION
RMS_RELATED_CONTEXT=NO
RESD$:[TMPOBJ]A.OBJ, RESD$:[SRC]B.OBJ, RESD$:[SRC]C, DSKD$:[OBJ]D.OBJ
DSKD$:[OBJ]E, RESD$:[TMP SRC]F.OBJ, RESD$:[TEST]G.OBJ
RESD$:[SRC.OBJ]H, RESD$:[COM]DOES_NOT_EXIST.OBJ Ctrl/Z

%LINK-F-OPENIN, error opening DISK_READ$:[SYS.][COM]DOES_NOT_EXIST.OBJ; as input
-RMS-E-FNF, file not found
$
```

5.24.3 ライブラリ・チェックにおける Linker のデフォルト動作の変更

V7.3-1

これまでの Linker では、ライブラリと共有イメージ間の一致条件が厳密に検証されていましたが（正確な日時を照合し、該当するものがない場合は、LINK-I-DATMISMCH シグナル通知を発行）、このリリースでは、イメージ・アクティベータと同じ検証（GSMATCH 条件を使用して互換性を検証）だけが実行されます。

以前の動作（日時の照合）を実行する場合は、LINK\$SHR_DATE_CHECK 論理名を設定してください。

5.24.4 スタックの要素数は最大 25 に制限

永続的な制限事項

オブジェクト・ファイルを作成する開発者は、Linker の内部スタックの要素数が最大 25 に制限されていることに注意しなければなりません。どのような計算も、この制限の範囲内で実行しなければなりません。

5.25 OpenVMS I64 用 Linker ユーティリティ

V8.2

ここでは、I64 Linker ユーティリティに関する注意事項について説明します。

Alpha Linker に関する注意事項は、第 5.24 節を参照してください。

Alpha Linker と I64 Linker に共通に適用される注意事項は、第 5.24.1 項を参照してください。

5.25.1 I64 リンカと Alpha リンカの違い

V8.2

OpenVMS I64 Linker と OpenVMS Alpha Linker の違いに関する詳細な説明については、『HP OpenVMS Version 8.2 新機能説明書』を参照してください。『HP OpenVMS Version 8.2 新機能説明書』には、OpenVMS I64 Linker の新機能についての説明もあります。

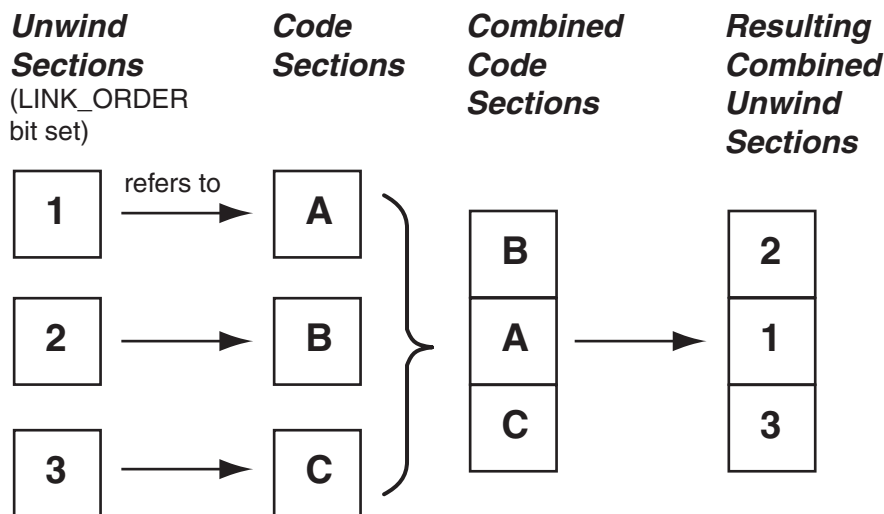
5.25.2 LINK_ORDER セクション・ヘッダ・フラグはサポートされていない

V8.2

Linker は、ELF セクション・ヘッダ・フラグ LINK_ORDER をサポートしていません。ただし、このフラグが設定されている unwind セクションは、正しい順序で配置されます。

このフラグは、セクションがイメージ・ファイル内の他のセクションと結合される場合、参照先のセクションが結合される順序と同じ順序で結合しなければならないことを示します。unwind セクションは、特別なケースとして扱われ、正しい順序で現れます。

次の図に、この概念を図示します。



VM-1134A-AI

5.25.3 data-reduced ELF オブジェクト・ライブラリとのリンクは推奨できない

V8.2

DCX data-reduced ELF オブジェクト・ライブラリは、ライブラリ内の連続領域には格納されません。その結果、最初にデータの展開を行う必要があるため、モジュールをプロセスの P2 空間に直接マッピングすることはできません。LBR\$MAP_MODULE ライブラリ・サービスが、モジュールを展開しプロセスの P2 空間にコピーします。この動作により、結果としてできた P2 空間内のページが、プロセス・クォータとしてカウントされます。

LBR\$UNMAP_MODULE ライブラリ・サービスは、これらのページを回復しますが、これらのページはヒープ領域解放リストに残り、プロセス・クォータとしてカウントされ続けます。そのため、Linker 操作の前に、あらかじめ DCX data-reduced ELF オブジェクト・ライブラリを展開することをお勧めします。

5.25.4 マップの Image Synopsis セクションの誤り

V8.2

マップの Image Synopsis セクションで、フィールド "Number of code references to shareable images:" と "Estimated map length:" の数値が正しくありません。この問題は、OpenVMS I64 の今後のリリースで修正される予定です。

5.25.5 DIFTYPE および RELODIFTYPE メッセージ

V8.2

OpenVMS I64 システムでは、あるモジュールでデータ (OBJECT) として定義した変数は、他のモジュールでもデータとして参照する必要があります。あるモジュールでプロシージャ (FUNC) として定義した変数は、他のモジュールでもプロシージャとして参照する必要があります。

データがプロシージャとして参照された場合、次のような情報メッセージが表示されます。

```
%ILINK-I-DIFTYPE, symbol symbol-name of type OBJECT cannot be  
referenced as type FUNC
```

プロシージャがデータとして参照された場合、次のような情報メッセージが表示されます。

```
%ILINK-I-DIFTYPE, symbol symbol-name of type FUNC cannot be  
referenced as type OBJECT
```

OpenVMS I64 では、リンカが関数記述子を生成するため、タイプのチェックはリンカで実施しています。OpenVMS Alpha では、同等のプロシージャ記述子をコンパイラが生成しているため、上記の情報メッセージは OpenVMS I64 のリンカで新しく追加されました。

プログラミングに関する注意事項

5.25 OpenVMS I64 用 Linker ユーティリティ

このメッセージは、通知のみなのでユーザ・アクションは必要としません。しかし、リンカは、データがプロシージャとして参照されたことを検出した場合、DIFTYPE メッセージに加えて、次のような警告メッセージを出力します。

```
%ILINK-W-RELODIFTYPE, relocation requests the linker to build a
function descriptor for a non-function type of symbol
```

下記の 2 つのモジュールの例は、これらの誤りを修正する方法を示します。

```
TYPE1.C
#include <stdio>

int status ; // Defines status as data.
extern int sub();

main ()
{
    printf ("Hello World\n");
    sub();
}
```

```
TYPE2.C

extern int status (int x) ;

sub ()
{
    int x;
    x = (int)status;
    return status (x);
}
```

これらのモジュールをリンクすると、次のような情報メッセージと警告メッセージが出力されます。

```
$ CC/EXTERN_MODEL=STRICT_REFDEF TYPE1
$ CC/EXTERN_MODEL=STRICT_REFDEF TYPE2
$ LINK TYPE1,TYPE2
%ILINK-I-DIFTYPE, symbol STATUS of type OBJECT cannot be referenced as
type FUNC
    module: TYPE2
    file: NODE1$:[SMITH]TYPE2.OBJ;6
%ILINK-W-RELODIFTYPE, relocation requests the linker to build a
function descriptor for a non-function type of symbol
    symbol: STATUS
    relocation section: .rela$CODE$ (section header entry: 18)
    relocation type: RELAS$K_R_IA_64_LTOFF_FPTR22
    relocation entry: 0
    module: TYPE2
    file: NODE1$:[SMITH]TYPE2.OBJ;6
```

問題を修正して、情報メッセージと警告メッセージが出力されないようにするには、TYPE1.C を修正して、status をプロシージャとして定義します。


```
TYPE1.C

#include <stdio>

int status (int x); // Defines status as a procedure.
extern int sub();

main ()
{
    printf ("Hello World\n");
    sub();
}

int status (int x) {
    return 1;
}

$ CC/EXTERN_MODEL=STRICT_REFDEF TYPE1
$ CC/EXTERN_MODEL=STRICT_REFDEF TYPE2
$ LINK TYPE1,TYPE2
```

5.26 LTDRIVER: CANCEL SELECTIVE の制限事項

永続的な制限事項

OpenVMS Version 6.1 より前のリリースでは、LTDRIVER は「拡張 DDT」ビットをセットしていませんでした。したがって、POSIX 関数 CANCEL SELECTIVE は LTDRIVER で動作しませんでした。この問題は解決されましたが、まだ制限事項が残っています。

この修正により、\$QIO 読み込みと書き込みを選択的に取り消すことができるようになりましたが、ポート・ドライバに対して行った \$QIO (つまり、LAT 接続 \$QIO などのように IO\$_TTY_PORT 関数修飾子を使用して行ったもの) は、CANCEL SELECTIVE によって取り消すことができません。

5.27 Mail ユーティリティ: 呼び出し可能メールのスレッドの制限事項

V7.1

OpenVMS 呼び出し可能メール・ルーチンはスレッド・セーフではありません。スレッド化されたアプリケーション内での非スレッド・セーフ・ルーチンの呼び出しの詳細については、『Guide to the POSIX Threads Library』を参照してください。

呼び出し可能メールのコンテキスト情報は、プロセス単位 (スレッド単位ではない) で管理されるので、コンテキスト・ベースの処理を実行する複数のスレッドは相互に同期をとり、特定のタイプのメール・コンテキストが一度に1つだけアクティブになるようにしなければなりません。この条件が満たされないと、1つのスレッドが他のスレッドのメール操作を妨害する可能性があります。

OpenVMS Alpha システムでは、マルチスレッド環境でカーネル・スレッドが有効に設定されている場合、この他にも追加制限事項があります。この環境では、呼び出し可能メールは初期スレッドでのみ使用しなければなりません。

5.28 OpenVMS Debugger

以降の注意事項では、OpenVMS Alpha システムと OpenVMS I64 システムでの、OpenVMS Debugger に固有の注意事項について説明しています。

Delta デバッガと XDelta デバッガの注意事項については、第 5.13 節を参照してください。System Code Debugger (SCD) の注意事項については、第 5.37 節を参照してください。

5.28.1 全般的な問題点と回避方法 (I64 のみ)

V8.2

次に、確認されている全般的な状況と、ユーザの回避方法を示します。

問題点: 要素が境界にアラインされていない配列は、正しく表示されません。これは、プログラミング言語の構文を使用して、不自然なデータ・アラインメントを強制した場合に発生します (たとえば、C 言語での `#pragma nomember_align`、Pascal での `PACKED` キーワード)。

回避方法: ありません。

問題点: 添字がスカラ (整数) の配列を調べるとき、デバッガは配列のインデックス値を 10 進数ではなく、現在定義されている基数で出力します。

回避方法: `SET RADIX DECIMAL` コマンドを実行します。

問題点: I64 システムの OpenVMS Debugger は、あたかもレジスタ・リネーム・ベース (CFM.rrb) とローテート・サイズ (CFM.sor) がいずれも 0 であるかのように、汎用レジスタ、浮動小数点レジスタ、およびプレディケート・レジスタを表示します。言い換えると、ローテートするレジスタを使用しているときは、ローテーションの効果は無視されます。この問題は、今後のリリースで修正される予定です。さらに詳細な説明は第 5.9 節を参照してください。

注意

この現象は、C++ およびアセンブラ言語のプログラムの、特殊な状況でまれに発生します。多くのプログラムは、この問題の影響を受けません。

回避方法: CFM レジスタを調べ、(ゼロでない) CFM.rrb フィールドと CFM.sor フィールドの値を考慮して、手作業で EXAMINE コマンドを調整します。

5.28.2 Basic 言語での問題 (I64 のみ)

V8.2

問題点: 多次元配列の範囲を調べるとき、デバッガは要素のアドレスを適切にシンボリ化できません。つまり、配列の行と列の値が正しくありません。

回避方法: ありません。

5.28.3 C++ 言語での問題 (I64 のみ)

V8.2

問題点: SHOW CALLS コマンドが、C++ のマングル化されていない名前ではなく、マングル化された名前を表示することがあります。

回避方法: ありません。

5.28.4 COBOL 言語での問題 (I64 のみ)

V8.2

問題点: EDIT picture 節で宣言したデータは、部分的にしかサポートされません。EXAMINE コマンドを使用して項目を調べると、位取りなどの調整なしに生データが表示されます。整数や 10 進数の定数を保存することで項目を変更しようとする (DEPOSIT コマンドを使用) とエラーになります。

回避方法: 適切な位取り因数を設定し、表示される値を手作業で調整します。値を変更する必要がある場合は、別の変数または引用符付きの文字列をソースとして DEPOSIT コマンドを実行します。

5.28.5 Fortran 言語での問題 (I64 のみ)

V8.2

問題点: REAL*16 型のデータ項目に値を保存すると、実際の値ではなく常にゼロが格納されます。

回避方法: ありません。

5.28.6 Pascal 言語での問題 (I64 のみ)

V8.2

問題点:動的境界で宣言された配列(たとえば "array [1..n]")が、正しく処理されないことがあります。配列を調べている際に、デバッガが INVARRDSC や IVALOUTBNDS のエラーを表示することがあり、SHOW SYMBOL/TYPE で配列境界が正しく表示されないことがあります。

回避方法:ありません。

5.28.7 SET SCOPE コマンド: 動作の変更

V8.2

SET SCOPE コマンドの動作は、このリリースで変更されました。SET SCOPE では、デバッガの言語設定を、指定されたスコープの言語に変更するようになりました。以前は、SET SCOPE は言語設定に影響を与えませんでした。SET LANGUAGE NODYNAMIC コマンドを実行することで、この機能を無効にできます。

5.28.8 SHOW IMAGE コマンドの変更

V8.2

SHOW IMAGE コマンドは、プロセスのイメージが占めるアドレス空間の概要を表示するようになりました。イメージ・セクション情報をすべて表示するには、SHOW IMAGE/FULL を使用するか、イメージ名を指定します(たとえば、SHOW IMAGE FOO)。

5.28.9 クライアント/サーバ・インタフェース: 以前のバージョンはサポートされない (Alpha のみ)

V7.3

OpenVMS Alpha Version 7.3 およびそれ以降の Debugger では、以前のバージョンのクライアント/サーバ・インタフェースをサポートしません。次の表に従って、配布メディアのキットにあるクライアント/サーバ・インタフェースをインストールしてください。

CPU	オペレーティング・システム	クライアント・キット
Intel	Microsoft Windows 95 , 98 , NT , Me , 2000 , XP	[DEBUG_CLIENTS011.KIT] DEBUGX86011.EXE
Alpha	Microsoft Windows NT	[DEBUG_CLIENTS011.KIT] DEBUGALPHA011.EXE

これらのクライアント・キットは、自己解凍形式の.EXE ファイルです。

適切な実行形式ファイルを PC に転送したら、そのファイルを実行して、PC 上にデバッグ・クライアントをインストールすることができます。InstallShield インストール・プロシージャのガイドに従ってインストールを行います。

5.29 OpenVMS のシステム・ダンプ・アナライザ (SDA)

ここでは、システム・ダンプ・アナライザ (SDA) に関する注意事項について説明します。

5.29.1 READ コマンドのデフォルトの変更

V8.2

SDA READ コマンドのデフォルトの動作は、/LOG から/NOLOG に変更されました。

5.29.2 CLUE コマンドは OpenVMS I64 に移植されていない

V8.2

次の CLUE コマンドは、OpenVMS I64 にはまだ移植されていないため、その旨のメッセージを返します。

```
CLUE CALL
CLUE ERRLOG
CLUE FRU
CLUE REGISTER
```

5.29.3 SHOW CALL_FRAME 機能の変更 (I64 のみ)

V8.2

OpenVMS I64 Version 8.2 で、次の機能変更が実施されました。

- 以下のコマンドで/ALL および/SUMMARY 修飾子が使用できるようになりました。
 - SHOW CALL *address*
 - SHOW CALL /HANDLE=*address*
 - SHOW CALL /EXCEPTION_FRAME=*address*
- SHOW CALL *address*で参照するアドレスは常に Invocation Handle アドレスでしたが、Exception Frame アドレスも参照できるようになりました。

*address*が Exception Frame アドレスの場合、そのフレーム・アドレスは、それ以降の呼び出しスタックをたどる際の起点として SDA に記憶されます。

*address*が、以降の SHOW CALL コマンドの Invocation Handle アドレスの場合、SDA は呼び出しスタックをたどる際に、その前に記憶した起点から始めます。

後続の*address*が、別の Exception Frame アドレスの場合、SDA は以前に記憶した呼び出しスタックの起点を廃棄し、新しい例外フレーム・アドレスを呼び出しスタックの起点として記憶します。

同様に、SHOW CALL/EXCEPTION_FRAME コマンドは、そこで指定された例外フレーム・アドレスを呼び出しスタックの起点とします。

この方法により、任意の例外フレームから呼び出しスタックをたどることができます。起点の例外フレームがシステム空間に存在しても、呼び出しチェーンの一部はプロセス空間に存在する可能性があるので注意してください(逆の場合もそのようになります)。いずれの場合も、すべての呼び出しスタックをたどるには、SDA の現在のプロセス・コンテキストは、呼び出しスタックの起点となっているプロセスのものであることが必要です。
- アドレス指定なしで、また/NEXT、/EXCEPTION_FRAME、/HANDLE 修飾子もなしで SHOW CALL コマンド、SHOW CALL/SUMMARY コマンド、SHOW CALL/ALL コマンドを入力すると、起点は、現在のプロセスのデフォルトの呼び出しチェーンにリセットされます。

5.30 OpenVMS I64 Version 8.2 に含まれない PL/I ライブラリ

V8.2

PL/I ライブラリ (ネイティブ・ライブラリと変換されたライブラリ) は、OpenVMS Alpha には含まれていましたが OpenVMS I64 には含まれていません。PL/I コア・イメージのうち OpenVMS I64 がないものは以下のとおりです。

```
[SYSLIB]DPLI$RTL$SHR.EXE
[SYSMMSG]PLI$MSG.EXE
[SYSLIB]PLIRTL.IIF
[SYSLIB]PLIRTL_D56_TV.EXE
```

PL/I ライブラリを参照するアプリケーションや共有イメージは、Integrity サーバ用の OpenVMS I64 では動作しません。(一般的に、PL/I で記述したコードを含むアプリケーションや共有イメージがそうです。)この制約は、ネイティブ・コードにも、変換された VAX および Alpha イメージにも、適用されます。

第 2.8 節の、関連する注意事項を参照してください。

5.31 POSIX スレッド・ライブラリ

ここでは、POSIX スレッド・ライブラリ(旧名称は、DECthreads)に関する注意事項について説明します。

第 A.8 節の関連する注意事項も参照してください。

5.31.1 例外処理中のスタック・オーバフロー (I64 のみ)

V8.2

I64 システムでの例外処理には、Alpha システムの場合よりもかなり大きなスタック領域が必要です。OpenVMS からアプリケーションを移植するときに、例外処理を使用するスレッドに未使用スタック領域が十分ないと、I64 での例外処理中に、このスレッドでスタック・オーバフローが発生することがあります。通常は、次の操作のいずれかに関連する ACCVIO の処理が不適切であったように見えます。

- RAISE
- pthread_cancel
- pthread_exit
- スレッドにアクティブな TRY または pthread_cleanup_push ブロックがあり、OpenVMS の状態がシグナル通知された(たとえば、ハードウェア例外や、LIB\$SIGNAL または LIB\$STOP の使用など)。

このような問題が発生した場合は、スレッドに割り当てられるスタックのサイズを数ページずつ増やしてみてください。最初は、スタックのサイズを 24 KB 大きくすることをお勧めします。

デフォルトのスタック・サイズは、I64 システム上でのスタック使用量が多いことに対応するため、24 KB 大きくされました。アプリケーションが多数のスレッドをデフォルトのスタック・サイズで作成している場合、アプリケーションのメモリ・リソースが不足することがあります。このような状況になった場合は、プロセス・クォータを大きくするか、アプリケーションを変更して同時に存在するスレッドの数を減らしてください。

5.31.2 I64 システムでの THREADCP コマンドの動作

V8.2

OpenVMS I64 システム上の DCL コマンド THREADCP は、スレッド関連の 2 つのメイン・イメージ・ヘッダ・フラグ、UPCALLS と MULTIPLE_KERNEL_THREADS の問い合わせや変更には使用できません。代わりに、I64 システムでのスレッド・ヘッダ・フラグの設定や参照を行うための DCL コマンド SET IMAGE および SHOW IMAGE を使用する必要があります。

Alpha システムのユーザは、引き続き THREADCP コマンドを使用してください。

5.31.3 浮動小数点のコンパイルと例外 (I64 のみ)

V8.2

次の 2 つの古い cma スレッド・ライブラリ・ルーチンのいずれかを呼び出すソース・モジュールは、OpenVMS I64 上で使用するために/FLOAT=G_FLOAT コンパイラ修飾子 (または、言語固有の同等の修飾子) を指定してコンパイルしなければなりません。

```
cma_delay()  
cma_time_get_expiration()
```

これらのルーチンは、VAX 形式の浮動小数点数だけを引数として受け入れます。通常、OpenVMS I64 コンパイラは、デフォルトで VAX 形式を使用する OpenVMS Alpha コンパイラとは異なり、デフォルトで IEEE 形式の浮動小数点数を使用します。この 2 つの cma スレッド・ルーチンは、Alpha と I64 のどちらでも、VAX 形式の浮動小数点引数だけを受け入れます。これらのルーチンの呼び出しを適切にコンパイルしないと、IEEE 形式の浮動小数点数が実行時に誤って渡され、予期しない例外が発生することがあります。

5.31.4 C 言語コンパイル・ヘッダ・ファイルの変更

V7.3-2

OpenVMS Version 7.3-2 では、次の変更が行われました。

- INTS.H

OpenVMS の以前の一部のリリースでは、POSIX スレッドの C 言語ヘッダ・ファイル PTHREAD_EXCEPTION.H に、C ヘッダ・ファイル INTS.H の #include が誤って含まれていました。この問題は、OpenVMS Version 7.3-2 で修正されました。PTHREAD_EXCEPTION.H を使用しても、コンパイルで INTS.H がインクルードされなくなりました。アプリケーションによっては、コンパイル時に

INTS.H が必要で、PTHREAD_EXCEPTION.H でインクルードされることを誤って前提としているものがあるかもしれません。

このようなアプリケーション・ソース・モジュールを OpenVMS Version 7.3-2 システムで再コンパイルすると、C コンパイラから診断メッセージが出力されません。これらのメッセージは、INTS.H で定義されるシンボルやデータ型（たとえば int32）が未定義であることを示します。このようなアプリケーション・ソース・モジュールを修正するには、該当するシンボルや型を最初に使用している位置より前で、直接<ints.h>を#include します。

- timespec_t typedef

以前のリリースの OpenVMS では、POSIX スレッドの C 言語ヘッダ・ファイル PTHREAD.H に、timespec_t という名前の typedef 定義が入っていました。このシンボルは、PTHREAD.H に属さない、非標準のシンボルです。（この typedef は、TIME.H や TIMERS.H などの C RTL ヘッダ・ファイルの内容との関連で残されていました。）OpenVMS Version 7.3-2 では、C RTL とスレッド・ヘッダ・ファイルの標準への準拠性が改善されました。この結果、PTHREAD.H から timespec_t の typedef が削除されました。

アプリケーションによっては、timespec_t の typedef がコンパイル時に必要で、PTHREAD.H で直接的または間接的（たとえば、TIS.H の使用による）に定義されることを誤って前提としているものがあるかもしれません。このようなアプリケーション・ソース・モジュールを OpenVMS Version 7.3-2 システムで再コンパイルすると、C コンパイラから、timespec_t を未定義のシンボルまたは型としてリストする診断メッセージが出力されます。このようなアプリケーション・ソース・モジュールを修正するには、timespec_t を使用している部分を timespec 構造体に置き換えるか、timespec_t シンボルを最初に使用している位置より前で、C RTL ヘッダ・ファイル TIMERS.H をインクルードします。

アプリケーションの構築環境で古い C RTL やスレッド・ヘッダ・ファイルのプライベート・コピーを使用していたり、timespec 構造体や timespec_t typedef を含む抜粋を使用している場合（どちらの方法もお勧めできません）は、もっと多くのコンパイル・エラーが表示されることがあります。コンパイラは、timespec 構造体が再定義されている（2 回定義されている）ことを示すメッセージを表示することがあります。このような場合、システムが提供している C RTL およびスレッド・ヘッダ・ファイルを使用するように戻したり、timespec 構造体に関連して個別に抜粋した箇所を、システムが提供している TIME.H ヘッダ・ファイルのインクルードに変更します。

5.31.5 新しい優先順位調整アルゴリズム

V7.3-2

OpenVMS Version 7.3-2 では、『Guide to the POSIX Threads Library』で説明されている適応型スレッド・スケジューリング動作が、新しい優先順位調整アルゴリズムとともに実装されました。場合によっては、新しいアルゴリズムでは、優先順位が

異なる、スループット方針のスレッドが同期オブジェクトを共用することによる問題を回避できます。優先順位の調整により、アプリケーションのスループットや、システム全体の使用状況も改善できます。スループット・スケジューリング方針のスレッドの優先順位調整は、自動で、透過的に行われます。

5.31.6 プロセス・ダンプ

V7.3

POSIX スレッド・ライブラリで実行時に修正不能な重大エラー (アプリケーション内のデータ破損によって損傷したデータ構造など) が検出されると、ライブラリにより実行中のイメージが終了されることがあります。終了中に、ライブラリによりプロセス・ダンプ・ファイルの作成がトリガーされます (このファイルは、ANALYZE/PROCESS_DUMP によりエラー診断に使用されます)。このようなプロセス・ダンプ・ファイルのサイズは、エラー時のプロセスのアドレス空間に依存するため、非常に大きくなる場合があります。

5.31.7 動的 CPU 構成の変更

V7.3

OpenVMS Version 7.3 以降、POSIX スレッド・ライブラリは、マルチプロセッサ Alpha システムを実行する CPU の数の動的変化に対応するようになりました。1 つのイメージに対して、複数のカーネル・スレッドが使用できるように指定 (LINK /THREADS_ENABLE 修飾子または THREADCP コマンド動詞により) すると、POSIX スレッド・ライブラリが、アプリケーションの明白な並列処理を監視して、利用可能な CPU の数を最大とする数のカーネル・スレッドを作成します。それぞれのカーネルスレッドは、OpenVMS エグゼクティブによってスケジューリングされて別々の CPU で実行されるので、同時に実行することができます。

アプリケーションの実行中、オペレータは CPU を個別に停止または開始することができます。このような動的変化を反映して、これ以降にイメージがアクティブ化されたときに作成できるカーネル・スレッドの数が増えます。また、現在実行中のイメージにも反映されるようになりました。

CPU を追加または除去すると、スレッド・ライブラリは、追加、除去後のアクティブな CPU の数を照会し、プロセスが現在使用しているカーネル・スレッドの数と比較します。現在 CPU がカーネル・スレッドよりも多い場合、ライブラリは既存の POSIX スレッドを CPU まで延長します (必要に応じて、すぐに、または後に新しいカーネル・スレッドを作成します)。逆に CPU がカーネル・スレッドよりも少ない場合、ライブラリは余分のカーネル・スレッドを強制的にハイバネートさせ、残りのカーネル・スレッド上で POSIX スレッドを再度スケジューリングします。これにより、プロセスに関する限り、利用可能な数以上のカーネル・スレッドが、CPU リソースを奪い合うということがなくなります。

5.31.8 デバッガ計測機能は動作しない

V7.0

POSIX スレッド・デバッガの計測機能は動作しません。

『Guide to the POSIX Threads Library』の C.1.1 に記載されている、動作中のプログラムをデバッグする手順を使用すると、プロセスが ACCVIO メッセージで失敗する可能性があります。

5.32 RMS: グローバル・バッファ VA 管理の改善

V8.2

以前のリリースでは、アプリケーションが RMS グローバル・バッファを使用してファイルのオープンとクローズをあるパターンで繰り返すと、VASFUL (仮想アドレス空間満杯) エラーとなり、アプリケーションが失敗することがありました。

OpenVMS Version 8.2 からは、RMS が拡張仮想アドレス (VA) 空間管理を持つようになったため、これらの VASFUL エラーが発生する可能性が大幅に減少しました。

5.33 RMS Journaling

ここでは、RMS Journaling for OpenVMS に関する注意事項について説明します。

RMS Journaling の詳細は、『RMS Journaling for OpenVMS Manual』を参照してください。このマニュアルは、OpenVMS Documentation CD-ROM (アーカイブ・マニュアルのディレクトリ) に入っています。

5.33.1 カーネル・スレッドと互換性のないリカバリ・ユニット・ジャーナリング

V7.3

DECdtm Services は複数カーネル・スレッド環境でサポートされず、RMS リカバリ・ユニット・ジャーナリングは DECdtm Services に依存しているため、RMS リカバリ・ユニット・ジャーナリングは、複数カーネル・スレッドが有効になっているプロセスではサポートされません。

5.33.2 ジャーナル・ファイル作成の変更

V7.2

Version 7.2 より前には、リカバリ・ユニット (RU) ジャーナルは、ジャーナリングされるファイルと同じボリュームの[SYSJNL]ディレクトリに一時的に作成されていました。リカバリ・ユニット・ジャーナルのファイル名は RMS\$*process_id* (*process_id*はプロセス ID の 16 進表現) という形式であり、ファイル・タイプは RMS\$JOURNAL でした。

OpenVMS Version 7.2 では、RU ジャーナル・ファイルの作成に関して、次の点が変更されました。

- ファイルは、[SYSJNL]ディレクトリのノード固有のサブディレクトリに作成される。
- リカバリ・ユニット・ジャーナルのファイル名は、YYYYYYYYY という形式に短縮された。ただし、YYYYYYYYY は逆の順にプロセス ID を 16 進数で表現したものです。

これらの変更により、ジャーナル・ファイルの作成と削除で発生するディレクトリのオーバーヘッドが減少します。

次の例に、以前のバージョンと現在のバージョンの両方のジャーナル・ファイルの作成を示します。

以前のバージョン: [SYSJNL]RMS\$214003BC.RMS\$JOURNAL;1
現在のバージョン: [SYSJNL.NODE1]CB300412.;1

RMS が[SYSJNL]ディレクトリまたはノード固有のディレクトリを見つけない場合は、RMS は自動的にそのディレクトリを作成します。

5.33.3 OSI 環境でのリカバリ・ユニット・ジャーナリングされたファイルへのリモート・アクセス

V6.1

ネットワーク内の他のノードからリモート・アクセスされるリカバリ・ユニット・ジャーナリング・ファイルのホストである OSI ノードでは、SYS\$NODE をフェーズ IV 形式のノード名として定義しなければなりません。SYS\$NODE によって指定されるノード名は、ホスト・ノードのリカバリ・ユニット・ジャーナリング・ファイルにアクセスしようとするすべてのリモート・ノードから認識されなければなりません。また、リモート・ノードがこのノード名を使用して、ホスト・ノードとの間で DECnet 接続を確立できるように、ノード名は固有の名前でなければなりません。この制限は、OSI または複合 OSI 環境と非 OSI 環境でネットワークを介してアクセスされる、リカバリ・ユニット・ジャーナリング・ファイルにだけ適用されます。

5.33.4 順方向 (AI) ジャーナリング

V6.0

順方向 (AI) ジャーナリングを使用すれば、使用不能またはアクセス不能になったデータ・ファイルを回復することができます。AI リカバリでは、AI ジャーナル・ファイルを使用して、データ・ファイルのバックアップ・コピーをロール・フォワードすることで、障害が発生した時点でのデータ・ファイルの新しいコピーが作成されます。

プロセスが削除されたりシステム障害が発生した場合には、更新情報を AI ジャーナル・ファイルに書き込むことができますが、データ・ファイルに書き込むことはできません。AI ジャーナリングだけが使用されている場合は、データ・ファイルとジャーナルの一貫性は自動的に維持されません。データ・ファイルに対して追加更新を行い、AI ジャーナルに記録すると、その後のロール・フォワード操作で一貫性のないデータ・ファイルが作成されることがあります。

リカバリ・ユニット (RU) ジャーナリングを AI ジャーナリングと組み合わせて使用した場合には、自動的なトランザクション・リカバリにより、AI ジャーナルとデータ・ファイルの間の一貫性が復元されます。

特定の状況では、AI ジャーナリングだけを使用するアプリケーションは、プロセスの削除やシステム障害の後でデータの不整合が発生しないように、予防措置をとることができます。たとえば、AI ジャーナリングされているファイルの手動ロール・フォワードを行うと、非共有 AI アプリケーション (シングル・アクセッサ) やスタンドアロン・システムで実行中の共有 AI アプリケーションなどが関連するシステム障害の発生後に、ファイルの一貫性を維持できます。

しかし、共有 AI アプリケーションでは、クラスタ内でプロセスの削除やシステム障害が発生した後で、AI ジャーナル・ファイルと同期のとれていないデータ・ファイルに対してこれ以上の操作が実行されないようにするための措置はとられません。このような状況では、データ・ファイルと AI ジャーナル・ファイルの間の一貫性は、AI ジャーナリングと RU ジャーナリングを組み合わせて使用することで維持できません。

5.33.5 VFC 形式の順編成ファイル

VAX V5.0

Alpha V1.0

逆方向ジャーナリングやリカバリ・ユニット・ジャーナリングを使用している場合、固定長制御部付可変長 (VFC) 順編成ファイルを更新することはできません。VFC 順編成ファイル形式は、FAB の FAB\$B_RFM フィールドのシンボリック値 FAB\$C_VFC によって示されます。

5.34 RTL ライブラリ (LIB\$)

ここでは、LIB\$ランタイム・ライブラリに関する注意事項について説明します。

5.34.1 RTL ライブラリ (LIB\$) のヘルプ

V8.2

OpenVMS Version 8.2 の LIB\$ランタイム・ライブラリのヘルプ・ファイルには、LIB\$LOCK_IMAGE ルーチンのヘルプがありません。この問題は、今後のリリースで修正される予定です。当面は、このルーチンの詳細な説明は『OpenVMS RTL Library (LIB\$) Manual』を参照してください。

5.34.2 RTL Library (LIB\$): 呼び出し標準ルーチン (I64 のみ)

V8.2

この注意事項では、ローテートするレジスタが、以下の呼び出し標準ルーチンでどのように取り扱われるかを明確化します。

```
LIB$I64_GET_FR  
LIB$I64_SET_FR  
LIB$I64_GET_GR  
LIB$I64_SET_GR  
LIB$I64_PUT_INVO_REGISTERS
```

呼び出し標準規則の ICB (invocation context block) およびメカニズム・ベクタは常に、あたかも、レジスタ・リネーム・ベース (CFM.rrb) とローテート・サイズ (CFM.sor) がいずれも 0 であったかのように、汎用レジスタ、浮動小数点レジスタ、およびプレディケート・レジスタを記録しています。言い換えると、ローテートするレジスタを使用しているときに、ローテーションの効果が無視されます。このことは、LIB\$I64_PUT_INVO_REGISTERS ルーチンが使用するレジスタ・マスクについても同様です。というのは、これらのマスクは、ICB 構造体のフィールドによって定義されるからです。

現在は、補足的なアクセス・ルーチン LIB\$I64_GET_FR、LIB\$I64_SET_FR、LIB\$I64_GET_GR および LIB\$I64_SET_GR が、レジスタ・リネーム・ベース・レジスタとローテート・サイズ・レジスタの効果を調整しないで、不適切に、そのレジスタ番号パラメータを解釈しています。これは、誤りであり今後のリリースで修正される予定です。

それまでは、ICB またはメカニズム・ベクタ内の汎用レジスタ、浮動小数点レジスタ、およびプレディケート・レジスタを調べるプログラムや、実行時に見えるレジスタを探して内容を解釈するプログラムでは、保存された CFM レジスタを調べて、自身で適切に調整する必要があります。

5.35 Screen Management (SMG\$) のドキュメント

『OpenVMS RTL Screen Management (SMG\$) Manual』の最後にある参照情報のトピックに、次の情報を追加します。

V7.2

- ルーチン SMG\$DELETE_VIRTUAL_DISPLAY の「Condition Values Returned (返される条件値)」に、次の説明を追加してください。

"Any condition value returned by the \$ DELPRC system service"
(\$DELPRC システム・サービスから返された条件値)

- ルーチン SMG\$GET_TERM_DATA の説明の「Arguments (引数)」の部分で、capability-data 引数の説明が誤っています。正しい説明は次のとおりです。

access: write-only
mechanism: by reference, array reference

- ルーチン SMG\$SET_OUT_OF_BAND_ASTS の説明の「引数 (AST-argument)」の部分で、AST-argument 引数の説明に誤りがあります。構造体の図のシンボル名が誤っています。この図の下にある段落に示されているシンボル名は正しい名前です。正しいシンボル名と誤ったシンボル名は次のとおりです。

誤っているシンボル名	正しいシンボル名
SMG\$L_PASTEBOARD_ID	SMG\$L_PBD_ID
SMG\$L_ARG	SMG\$L_USER_ARG
SMG\$B_CHARACTER	SMG\$B_CHAR

V7.1

- SMG\$READ_COMPOSED_LINE ルーチンの説明で、flags 引数の説明に次の文を追加してください。

"The terminal characteristic /LINE_EDITING should be set for your terminal for these flags to work as expected. /LINE_EDITING is the default."

(「これらのフラグが正しく機能するには、端末に対して端末属性/LINE_EDITING を設定しなければなりません。/LINE_EDITING は省略時の設定です。」)

- ルーチン SMG\$SET_KEYPAD_MODE の説明に、次の注意を追加してください。

注意

キーパッド・モードを変更すると、物理端末の設定も変更されます。これは、keyboard-id 引数によって指定される仮想キーボードだけでなく、すべての仮想キーボードに対するグローバルな変更です。

5.36 SORT32 ユーティリティ

ここでは、OpenVMS Alpha および OpenVMS I64 Version 8.2 用の、SORT32 V08-010 に関する注意事項について説明します。詳細は、第 5.20.8 項と第 5.20.1 項を参照してください。

Hypersort で修正されていない問題を回避する場合、または Hypersort に実装されていない機能を使用する場合に SORT32 を使用することをお勧めします。Hypersort の注意事項については、第 5.20 節を参照してください。

5.36.1 DFS サービス・ディスクでの CONVERT の問題

V8.2

SORT、MERGE、および CONVERT 操作は、UNIX がサービスする DFS マウント・ディスクが出力先になっている場合、%SORT-E-BAD_LRL エラーを返します。

この制約事項を回避するには、次のいずれかを実行します。

- 出力ファイルを OpenVMS ディスクに書き込んでから、その出力ファイルを UNIX がサービスする DFS マウントディスクにコピーする。
- CONVERT/FDL コマンドを使用する。このとき、FDL には、出力ファイルに必要な LRL (最大レコード長) を指定する。

5.36.2 一時作業ファイルが削除されないことがある

V7.3-2

SORT32 は、一時作業ファイルを削除しないことがあります。SYS\$SCRATCH や、SORT32 の作業ファイルを置いている場所を定期的にチェックし、削除されていない作業ファイルを削除してディスク・スペースを空けることができないかを調べてください。

5.36.3 複合条件のある SORT/SPECIFICATION: 要件

V7.3-1

SORT32 では、キー指定ファイルの複合条件が括弧で囲まれていない場合、複合条件に対する診断メッセージを出力しません。例を次に示します。

誤り:

```
/CONDITION=(NAME=TEST1, TEST=(Field2 EQ "X") AND (Field3 EQ "A"))
```


正しい:

```
/CONDITION=(NAME=TEST1, TEST=((Field2 EQ "X") AND (Field3 EQ "A")))
```

5.36.4 可変長レコードでの性能の問題

V7.3-1

SORT32 では、入力ファイル内の最大レコード長 (LRL) 情報に基づいて、ソート作業ファイルの固定長のスロットが割り当てられます。性能を向上させるには、実際の最大レコード長に最も近い LRL 情報を入力ファイルに設定します。初期性能が低い場合は、C プログラムによって作成されたファイルをソートしており、LRL が必要以上に大きく (32767 まで) 設定されていることが原因と考えられます。

5.36.5 作業ファイル・ディレクトリの制約事項

V7.3

SORT32 の作業ファイルは、必要な数の作業ファイルを複数のファイル・バージョンにわたって格納できるディレクトリに作成する必要があります。

5.37 System Code Debugger (SCD) はまだ I64 システムで使用できない

V8.2

『OpenVMS System Analysis Tools Manual』で System Code Debugger (SCD) は I64 および Alpha システムで使用できるとしてあります。実際には、SCD は Version 8.2 の I64 では使用できません。今後のリリースで使用可能になる予定です。

5.38 システム・サービス

ここでは、OpenVMS のシステム・サービスに関する注意事項について説明します。

5.38.1 \$GETJPI の項目コード SCHED_CLASS_NAME の記述誤り

V8.2

Version 8.2 の『OpenVMS System Services Reference Manual』とオンライン・ヘルプで \$GETJPI の項目コード SCHED_CLASS_NAME が、誤って CLASS_NAME と記載されています。この誤りは、次のメジャー・リリースで訂正される予定です。

5.38.2 PFN マップ・セクションに必要となる新しい GETSYI 項目コード (I64 のみ)

V8.2

OpenVMS VAX および Alpha では、PFN は 32 ビット値 (ビット 0 ~ 31) です。ビット 31 が設定されている場合は、キャッシュされない I/O 空間を示します。

OpenVMS I64 では、PFN は 64 ビット値です。PFN マップ・セクションを使用する、I64 用のプログラムは、大きなデータ・サイズを認識するように変更済みであることを示さなければなりません。これを示すには、次のシステム・サービスの呼び出しで、新しいフラグ SEC\$M_ARGS64 を設定します。

```
SYS$CRMPSC_PFN_64  
SYS$CREATE_GPFN  
SYS$CRMPSC_GPFN_64  
SYS$MGBLSC_GPFN_64
```

SEC\$M_ARGS64 フラグは OpenVMS Alpha では無視されますが、OpenVMS I64 では設定されていなければなりません。OpenVMS I64 でこのフラグが設定されていない場合、エラー・コード SS\$IVSECFLG が返されます。

32 ビット用のサービス SYS\$CRMPSC および SYS\$MGBLSC を使用して、OpenVMS I64 上の PFN マップ・セクションを作成したり、マッピングすることはできません。

OpenVMS I64 上の PFN の大きいデータ・サイズに対応するために、SYS\$GETSYI システム・サービスが、新しい項目コード SYI\$_PFN_MEMORY_MAP_64 を認識するように拡張されました。この項目コードの詳細については『OpenVMS System Services Reference Manual』を参照してください。

SYI\$_PFN_MEMORY_MAP_64 が指定されると、このサービスは物理メモリのレイアウトを、64 ビットのデータ構造体に返します。I64 システムでは、新しい項目コードと新しいレイアウトを使用しなければなりません。OpenVMS Alpha は、新しい形式を受け付けるように拡張されていますが、古い形式も引き続きサポートしています。

5.38.3 PFN マップ・セクションとキャッシュされないメモリ (I64 のみ)

V8.2

OpenVMS I64 システムでのマップされた I/O 空間には、キャッシュなしのアクセスを行わなければなりません。PFN マップ・セクションをキャッシュされないメモリとして扱う必要がある場合は、このセクションの作成時に SEC\$M_UNCACHED フラグを設定しなければなりません。次のシステム・サービスが、SEC\$M_UNCACHED フラグを受け入れるようになりました。

```
SYS$CRMPSC_PFN_64  
SYS$CREATE_GPFN  
SYS$CRMPSC_GPFN_64
```

SYS\$MGBLSC_GPFN_64 サービスは、このフラグを受け入れませんが、無視します。CACHED/UNCACHED 特性は、セクション属性として格納され、システムはセクションのマッピング時にこの属性を使用します。

OpenVMS Alpha システムでは、4つのサービスすべてが SEC\$M_UNCACHED フラグを受け入れませんが、無視します。

古いサービス SYS\$CRMPSC と SYS\$MGBLSC はアップデートされていないため、この新しいフラグは受け入れません。

5.38.4 SYS\$ACM: I64 システムでの使用

V8.2

OpenVMS I64 Version 8.2 システムでは、ACME_SERVER プロセスは手動で起動する必要があります。ACME_SERVER は SYS\$ACM 要求を処理します。SYS\$ACM システム・サービスを使用するアプリケーションがある場合は、SYS\$ACM を使用するアプリケーションを起動する前に、以下のコマンドを SYS\$STARTUP_VMS.COM プロシージャに追加してください。

```
$ SET SERVER ACME/START/LOG  
$ SET SERVER ACME/CONFIG=(NAME=VMS,CRED=VMS)  
$ SET SERVER ACME/ENABLE
```

注意

Advanced Server で提供される LAN Manager 認証を使用する場合は、必要な I64 コンポーネントの入手と設定手順について、Advanced Server V7.3A ECO4 のリリース・ノートを参照してください。

ACME_SERVER プロセスは、OpenVMS I64 の今後のリリースでは自動起動になる予定です。

5.38.5 I64 では SYS\$GOTO_UNWIND は利用できない

V8.2

SYS\$GOTO_UNWIND システム・サービスは、OpenVMS I64 システムでは利用できません。新しいサービス SYS\$GOTO_UNWIND_64 を使用するようにコーディングし直してください。

アプリケーションが正しく移植されるように、SYS\$GOTO_UNWIND は削除されました。詳細は、『HP OpenVMS Alpha から OpenVMS I64 へのアプリケーション・ポータリング・ガイド』を参照してください。

SYS\$GOTO_UNWIND_64 サービスは、OpenVMS Alpha Version 7.3-2 およびそれ以降にも存在します。このため、アプリケーションでは共通のコードを使用できます。SYS\$GOTO_UNWIND_64 の詳細は、オンライン・ヘルプ、または『OpenVMS System Services Reference Manual』を参照してください。

5.39 タイマ・キュー・エントリ (TQE)

V7.3-1

OpenVMS Alpha Version 7.3-1 では、タイマ・キュー・エントリの管理方法が変更され、多くの TQE を使用するシステムの性能が大きく向上しました。この変更は、非特権アプリケーションにとっては無関係です。

また、特権コードで TQE を直接操作することはできません。特に TQE キュー・ヘッダ (TQE\$L_TQFL/TQE\$L_TQBL) 内のポインタに直接アクセスすると、通常はアクセス違反になります。ただし、特権コードで内部ルーチン `exe_std$instimq` / `exe$instimq` と `exe_std$rmvtimq` / `exe$rmvtimq` を使用して、タイマ・キュー・エントリを入力または削除することは可能です。

5.40 Traceback 機能

ここでは、Traceback 機能 (TRACE) に関する注意事項について説明します。

5.40.1 API エラー (I64 のみ)

V8.2

ここでは、『HP OpenVMS Version 8.2 新機能説明書』で説明されている新しいトレースバック機能の API に関する注意事項について説明します。現在は、`rel_pc` (相対 PC 値) や `image_desc` (イメージ名記述子) の引数にゼロを指定すると、処理エラーになります。これらの引数がゼロの場合でも、トレースバック機能では、これらの引数を無視してトレースバック処理を継続する必要があります。ところがトレースバックでは、これらの引数がゼロだと、トレースバック処理を中断し、呼び出し元プログラムに失敗の戻り値を返します。この問題は、今後のリリースで修正される予定です。

5.40.2 問題の修正

V8.2

これまで、Traceback 機能は、/RESIDENT 修飾子または/SHARED=ADDRESS_DATA 修飾子を指定してインストールされたイメージを正しく処理しませんでした。この問題が修正されました。TRACE は、/RESIDENT 修飾子または/SHARED=ADDRESS_DATA 修飾子を指定してインストールされたイメージにシンボル情報が含まれていれば(つまり、/TRACEBACK を指定してリンクされていれば)、これらのイメージ内の位置をシンボルで表示するようになりました。

5.41 Watchpoint ユーティリティ (I64 のみ)

V8.2

Watchpoint ユーティリティは、まだ OpenVMS I64 に移植されていません。弊社では、このユーティリティを今後のリリースで移植する予定です。

5.42 プログラム全体の浮動小数点モード (I64 のみ)

V8.2

OpenVMS Alpha では、浮動小数点丸め動作、例外動作、および精度制御は、コンパイル時に定義されます。各モジュールは、それぞれの浮動小数点動作の設定で、個別にコンパイルされます。たとえば、計算のオーバフローでオーバフロー例外がシグナル通知されるディレクティブで1つのモジュールをコンパイルし、別のモジュールを、計算のオーバフローで例外をシグナル通知するのではなく、値を InfinityT とするディレクティブでコンパイルすることができます。これらの2つのモジュールがコンパイルされ実行された場合、モジュールのコードは、コンパイル時に指定されたオーバフロー動作をします。

OpenVMS I64 では、浮動小数点丸め動作、例外動作、および精度制御は実行時に定義され、プログラム全体の浮動小数点モードの概念で制御されます。プログラム全体の浮動小数点モードでは、プログラムのメイン・エントリ・ポイント(リンカが決定したもの)を含むモジュールが、デフォルトの浮動小数点丸め動作、例外動作、および精度制御を定義するモジュールです。

大半のプログラムには、この相違点の影響はありません。要点は、ホワイト・ペーパー『Intel® Itanium®アーキテクチャにおける OpenVMS 浮動小数点演算について』を参照してください。このドキュメントは、次の Web サイトで参照できます。

<http://h71000.www7.hp.com/openvms/integrity/resources.html>

ハードウェアに関する注意事項

この章では、以下のハードウェア製品についての情報を示します。

- MP コンソールと BMC コンソール (第 6.1 節)
- AlphaServer 2100 (第 6.2 節)
- AlphaServer 8200/8400 (第 6.3 節)
- AlphaServer ES47/ES80/GS1280 システム (第 6.4 節)
- AlphaServer GS シリーズ (第 6.5 節)
- AlphaStation 200/400 (第 6.6 節)
- AlphaStation 255 (第 6.7 節)
- ATI RADEON 7000 グラフィック (第 6.8 節)
- ATI RADEON 7500 グラフィック (第 6.9 節)
- DECwindows X11 ディスプレイ・サーバ (第 6.10 節)
- DIGITAL モジュラ・コンピューティング・コンポーネント (第 6.11 節)
- Digital パーソナル・ワークステーション (第 6.12 節)
- デュアル・コントローラ HSGnn デバイス (第 6.13 節)
- Open3D グラフィック (第 6.14 節)
- PowerStorm 300/350 PCI グラフィック・コントローラ (第 6.15 節)
- RFnn DSSI ディスク・デバイス (第 6.16 節)
- RZnn ディスク・デバイス (第 6.17 節)
- ZLX グラフィック・ボード (第 6.18 節)

デバイス・ドライバの使用についての注意事項も、この章の最後にまとめてあります。

6.1 MP コンソールと BMC コンソールの制約事項 (I64 のみ)

ここでは、MP コンソールと BMC コンソールに関する注意事項について説明します。

6.1.1 入力デバイス，出力デバイス，およびエラー・デバイスの制約事項

V8.2

現在，OpenVMS オペレーティング・システムでは，各 MP コンソールや BMC コンソールの入力デバイス，出力デバイス，およびエラー・デバイスは，同じシリアル回線コンソールに対応していなければなりません。システムに MP コンソールがある場合は，このコンソールを使用してください。

指定されたコンソールからブートしないと，VMS_LOADER に警告が送信され，ブート中にその他のエラーが表示されることがあります。また，通常のブート時に表示される情報が出力されないこともあります。

6.1.2 削除キーへの Ctrl/H の再マッピング

V8.2

文字 0X7F を DEL/RUBOUT 用に使用する OpenVMS オペレーティング・システムとは異なり，MP コンソール，BMC コンソール，および EFI コンソール環境では，Ctrl/H を使用します。VTxxx 端末上の削除キーを押したり，ターミナル・エミュレータで 0X7F を送信するようにマッピングされているキーを押しても，文字は削除されません。

注意: ドライバは，次の条件では再マッピングを行いません。

- 端末が IO\$_READALL 状態の場合
- 端末が IO\$_READPBLK 状態の場合
- 端末属性に PASSALL が設定されている場合
- 端末属性に PASTHRU が設定されている場合
- Ctrl/V が押されたことにより，ドライバが，次の文字を渡し，再マッピング・チェックを行わないようにされた場合

6.2 AlphaServer 2100

ここでは，AlphaServer 2100 シリーズのコンピュータ固有の情報をまとめます。

6.2.1 コンソール表示

V7.2

AlphaServer 2100 システムと 2100A システムで表示される次のようなコンソール表示は正常であり、システム・エラーを示しているわけではありません。

```
P00>>>SET CONSOLE SERIAL
P00>>>INIT

VMS PALcode X5.48-112, OSF PALcode X1.35-81

starting console on CPU 0
initialized idle PCB
initializing semaphores
initializing heap
initial heap 1c0c0
memory low limit = 132000
heap = 1c0c0, 13fc0
.
.
.
probing hose 0, PCI
probing PCI-to-EISA bridge, bus 1
probing PCI-to-PCI bridge, bus 2
*** unable to assign PCI base address
*** bus 2, slot 7, function 0, size 00001000 (16 bit I/O)
bus 1, slot 1 -- fra -- DEFEA
bus 1, slot 2 -- vga -- Compaq Qvision
bus 1, slot 3 -- pua -- KFESA
bus 2, slot 1 -- pka -- NCR 53C810
bus 2, slot 6 -- pkb -- NCR 53C810
bus 2, slot 7 -- pkc -- DEC KZPSA
bus 0, slot 7 -- ewa -- DECchip 21041-AA
initializing keyboard
Memory Testing and Configuration Status
Module  Size  Base Addr  Intlv Mode  Intlv Unit  Status
-----  ----  -
0       64MB  00000000   1-Way      0           Passed
Total Bad Pages 0
Testing the System
Testing the Disks (read only)
Testing the Network
econfig:      20041 99
econfig:      20042 04
econfig:      20043 00
AlphaServer 2100A Console V4.3-130, built on Oct 26 1996 at 19:44:57
P00>>>P
```

この表示では、KZPSA アダプタは正しくインストールされていますが、次のエラー・メッセージが表示されます。

```
*** unable to assign PCI base address
*** bus 2, slot 7, function 0, size 00001000 (16 bit I/O)
```

6.2.2 SCSI コントローラの制限事項

V6.2

1 GB 以上のメモリを装備した AlphaServer 2100 システムでは、Adaptec 1740 /1742 SCSI コントローラ (PB2HA-SA) はサポートされていません。コントローラがこのようなシステムに接続されていると、次のメッセージがオペレータのコンソールに表示されます。

```
%PKJDRVR-E- The direct DMA window does not map all of memory.  
Port is going OFFLINE.
```

6.3 AlphaServer 8200/8400: FRU テーブル・エラー

V7.2

エラー・ログ・バッファのサイズはシステム・パラメータ ERLBUFFERPAGES で制御され、最大値は 32 ページレットです。AlphaServer 8200/8400 または 4100 システムで、OpenVMS Alpha オペレーティング・システムのブート時に FRU (Field Replaceable Unit) テーブルがこの上限を超える場合には、エントリはエラー・ログ・ファイルに書き込まれません。

6.4 AlphaServer ES47/ES80/GS1280 システム

ここでは、AlphaServer ES47/ES80/GS1280 システムに関する注意事項について説明します。第 6.5.2 項の注意事項も、これらのシステムに関係します。

6.4.1 ES47/ES80/GS1280 ソフト・パーティションでの INIT コンソール・コマンドの使用

V8.2

同じハード・パーティション内で他のソフト・パーティションがブートされ、OpenVMS が実行されている場合には、ES47/ES80/GS1280 ソフト・パーティションでの INIT コンソール・コマンドの使用はサポートされません。INIT コマンドを実行すると、OpenVMS を実行中の他のインスタンスでシステム・クラッシュが発生することがあります。INIT コマンドを実行する前に、他のインスタンスをシャット・ダウンしてください。

コンソールの INIT の処理中に、同一ハード・パーティション内の他のインスタンスに対してブートコマンドを発行しないでください。INIT コマンドの完了までお待ちください。

6.4.2 SYSGEN パラメータ PHYSICAL_MEMORY の設定

V7.3-2

AlphaServer ES47/ES80/GS1280 システムのハードウェア構成要件上、システム・パラメータ PHYSICAL_MEMORY の設定を、デフォルトの -1 から変更することはお勧めできません。メモリ量を人為的に減らすと、システムで予期しない状況が発生することがあります。

6.4.3 RAD のサポート

V7.3-2

OpenVMS でのリソース・アフィニティ・ドメイン (RAD) のサポート (NUMA サポートまたは NUMA 対応とも言います) は、AlphaServer ES47/ES80/GS1280 システム用の OpenVMS Alpha Version 7.3-2 ではテストされていません。RAD サポートについての詳細は、『OpenVMS Alpha パーティショニングおよび Galaxy ガイド』を参照してください。

6.4.4 ライセンス要件

V7.3-2

AlphaServer ES47/ES80/GS1280 システムには、少なくとも 2 つの OpenVMS ソフトウェア・ライセンス (ベース・サポート用と、最初の 2 つのプロセッサのデュアル SMP サポート用) が必要です。この要件は、OpenVMS AlphaServer SMP システムの以前のライセンス付与方法から変更されています。OpenVMS のデュアル SMP ライセンスは、OpenVMS システムを購入したとき、または OpenVMS システムの追加 CPU モジュールを購入したときに、CPU モジュールに含まれています。

6.4.5 STOP/CPU およびシャットダウン動作

V7.3-2

ハードウェアの制約により、I/O ドロアを装備した AlphaServer ES47/ES80 /GS1280 システム上の CPU は、DCL コマンドの STOP/CPU を使用して停止することができません。一方、I/O ドロアを装備していないシステム上の CPU は、このコマンドで停止できます。

I/O ドロアを装備した ES47/ES80/GS1280 システムでシャットダウン・プロシージャを起動すると、次のようなエラー・メッセージが表示されることがあります。

```
%SYSTEM-W-WRONGSTATE, CPU 5 is in the wrong state for the requested operation
```

このメッセージは無視して構いません。シャットダウンは正常に実行されます。

6.4.6 MBM での時刻の設定

V7.3-2

AlphaServer ES47/ES80/GS1280 システムでは、MBM で正しい時刻と日付を設定しなければなりません。正しく設定しないと、OpenVMS は誤った日時を表示することがあります。

6.5 AlphaServer GS シリーズ・システム

ここでは、AlphaServer GS シリーズ・システムの大半のユーザに関する注意事項について説明します。

6.5.1 AlphaServer GS80/160/320 システム: デバイスの制限事項

永続的な制限事項

OpenVMS Alpha Version 7.3 以降では、1つのパーティションあたり、従来型のバス・アダプタ上のデバイスが1セットだけ設定され、サポートされます。このようなデバイスは、次のとおりです。

- シリアル・ポート COM1 および COM2
- パラレル・ポート
- キーボード
- マウス

従来型のバス・アダプタが複数ある場合には、コンソール・ポートを含むアダプタだけが設定され、サポートされます。

6.5.2 OpenVMS Galaxy ライセンスの実行

V7.3

OpenVMS Galaxy のコンピューティング環境では、システム・スタートアップ時やインスタンス間での CPU の再割り当てが発生するつど、OPENVMS-GALAXY ライセンス・ユニットがチェックされます。

CPU を起動しようとしたときに十分な OPENVMS-GALAXY ライセンス・ユニットがないと、CPU はインスタンスの設定されたセットの中に引き続き残りますが、動作は停止します。その後、適切なライセンス・ユニットをロードすれば、停止している CPU をシステムの実行中に起動することができます。これは、CPU が単独の場合でも複数の場合でも同じです。

6.5.3 ライセンスのインストール

V7.3-1

Version 7.3-1 以降にアップグレードする前に、次の手順を実行して、共通ライセンス・データベースが、ハード・パーティションとソフト・パーティション間でライセンス・ユニットを共有できるようにします。

1. 必要なユニット数を計算します。

- OpenVMS 基本ライセンスをロードする。
- SMP ライセンスをロードする。
- 次のコマンドを使用して、ライセンス・ユニットの数が適切か確認する。

```
$ SHOW LICENSE /UNIT_REQUIREMENTS /CLUSTER
```

注意

OpenVMS 基本ライセンスでは、パーティションではなく物理システムごとに、1人の操作ユーザのみを割り当てることができます。ただし、各パーティションの OPA0: からいつでもログインできます。操作ユーザを追加する場合は、追加のライセンス・ユニットが必要です。必要なライセンス・ユニット数については、弊社のサポート担当者にお問い合わせください。

2. 共通ライセンス・データベースにライセンスを追加します。たとえば、次のように入力します。

```
$ LICENSE REGISTER license-name /ISSUER=DEC -  
_$_ /AUTHORIZATION=USA123456 -  
_$_ /PRODUCER=DEC -  
_$_ /UNITS=1050 -  
_$_ /AVAILABILITY=H -  
_$_ /OPTIONS=(NO_SHARE) -  
_$_ /CHECKSUM=2-BGON-IAMA-GNOL-AIKO
```

LICENSE REGISTER コマンドで/INCLUDE 修飾子を使用して、ライセンスの NO_SHARE 属性を変更することはできません。

3. ライセンスを編集して、LICENSE REGISTER /INCLUDE=(node-name-list) コマンドで PAK の NO_SHARE 属性を変更します。たとえば、次のように入力します。

```
$ LICENSE MODIFY OPENVMS-ALPHA /INCLUDE=(NODEA, NODEB, NODEC)
```

4. 各パーティションで実行している OpenVMS のインスタンスで OpenVMS Alpha のライセンス・ユニットを利用できるようにするために、SRM 環境変数 SYS_

SERIAL_NUM が各パーティションで同じであるか確認します。確認するには、次の手順を実行します。

- a. 各パーティションのマスタ・コンソール(通常はコンソール・ライン 0)で、SHOW SYS_SERIAL_NUM コマンドを使用して、システムのシリアル番号を表示します。たとえば、次のように入力します。

```
P00>>>  
P00>>>SHOW SYS_SERIAL_NUM  
sys_serial_num G2A105
```

SYS_SERIAL_NUM の値がブランクの場合は、他の各パーティションのマスタ・コンソールで SHOW SYS_SERIAL_NUM コマンドを使用して、ブランクでないシステム・シリアル番号がないかチェックします。

注意

すべてのパーティション・コンソールに SYS_SERIAL_NUM の値がブランクで表示される場合は、0 以外の値を 12 文字以内で作成してください。作成するシステム・シリアル番号が、同じ OpenVMS Cluster の他の AlphaServer GS80/160/320 で使用されていないことを確認してください。

- b. システムのシリアル番号が決まったら、各パーティションのマスタ・コンソールで SET SYS_SERIAL_NUM コマンドを使用して、SYS_SERIAL_NUM の値を適切な値に変更します。たとえば、次のように変更します。

```
P00>>>  
P00>>>SET SYS_SERIAL_NUM G2A105
```

この手順は、すべてのハード・パーティションとソフト・パーティションで実行する必要があります。

5. OpenVMS Cluster ライセンス・データベースを正しく更新するために、すべての OpenVMS Cluster 共通ノードをシャットダウンしてからリブートすることをお勧めします。ローリング・アップグレード・タイプのブートでは、共通ライセンス・データベースは適切に更新されません。

注意

システムが、共通ライセンス・データベースを共有する OpenVMS Cluster の一部である場合、AlphaServer GS80/160/320 のハード・パーティションとソフト・パーティションの数を再構成する際には、必ずすべてのパーティションの SYS_SERIAL_NUM を同じ値にしてください。

パーティション間で NO_SHARE ライセンスを共有しているパーティション化可能なマシンでは、システムのブート時に次のエラー・テキストが表示される場合があります。

```
%LICENSE-E-NOAUTH, DEC OPENVMS-ALPHA use is not authorized on this node  
-LICENSE-F-EXCEEDED, attempted usage exceeds active license limits  
-LICENSE-I-SYSGR, please see your system manager  
Startup processing continuing...
```

このエラー・テキストは無視して構いません。このテキストは、OPENVMS-ALPHA PAK を共有しているシステムにユーザがログインして、使用中になった場合に表示されます。この問題は今後のリリースで修正される予定です。

6.5.4 AlphaServer GS60/GS60E/GS140 複数 I/O ポート・モジュール構成の制限事項

V7.2-1

複数の I/O ポート・モジュール (KFTHA-AA または KFTIA-AA) がある AlphaServer GS60/GS60E/GS140 構成では、システム障害が発生することがあります。

GS60/GS60E/GS140 システムの複数 I/O ポート・モジュールを持つ OpenVMS Galaxy および非 Galaxy の AlphaServer 8200/8400 構成をアップグレードするときには、Compaq Action Blitz # TD 2632 に記載されているように、最低でもリビジョン B02 KN7CG-AB EV6 CPU (E2063-DA/DB rev D01) のモジュールを 1 つインストールしなければなりません。

この制限事項と解決方法についての詳細は、Compaq Action Blitz # TD 2632 を参照してください。

6.6 AlphaStation 200/400: ISA_CONFIG.DAT の変更が必要

V7.1

AlphaStation 200/400 ファミリ・システムで ISA 装置を構成する場合には、各デバイスのノード情報が各デバイス記述ブロックの最後に格納されるように、SYS\$MANAGER:ISA_CONFIG.DAT ファイルを変更しなければなりません。

重要

OpenVMS Version 6.2 または 7.0 システムからアップグレードする場合は、アップグレード手順を開始する前に、この変更を行わなければなりません。

表 6-1 に、デバイス記述ブロックの変更点を示します。

表 6-1 デバイス記述ブロックの変更点

Version 7.1 より前	Version 7.1 以降
[AUA0]	[AUA0]
NAME=AU	NAME=AU
NODE=3	DRIVE=SYS\$MSBDRIVER
DRIVER=SYS\$MSBDRIVER	IRQ=9
IRQ=9	DMA=(0,1)
DMA=(0,1)	PORT=(388:4,530:8)
PORT=(388:4.530:8)	NODE=3

SYS\$MANAGER:ISA_CONFIG.DAT ファイルを使用している場合には、第 A.7 節を参照してください。

6.7 AlphaStation 255: PCI 構成の制限事項

V7.1

AlphaStation 255 シリーズ・システムの OpenVMS Alpha オペレーティング・システムでは、PCI スロット 0 に PCI オプション・カードを構成することはできません。

PCI スロット 0 は、AlphaStation 255 シリーズ・システム上の、最下位の物理 PCI オプション・スロットです。このスロットの割り込みシグナルは、組み込みイーサネット・ポートと共用されます。OpenVMS Alpha オペレーティング・システムは現在、PCI デバイスが割り込みラインを共用することを許していないため、スロット 0 に PCI デバイスを設置すると、正しく動作しないか、組み込みのイーサネット・ポートでエラーが発生する原因となることがあります。この制限事項があるため、AlphaStation 255 シリーズ・システムがサポートする PCI オプション・カードの数は、最大で 2 枚(スロット 1 とスロット 2 に構成)です。

6.8 ATI RADEON 7000 グラフィック (I64 のみ)

V8.2

ここでは、OpenVMS I64 システムで、組み込み型 ATI RADEON 7000 グラフィック・アダプタを使用する際の注意事項について説明します。

注意: 第 6.9.1 項で説明しているリソースの要件は、組み込み型 ATI RADEON 7000 グラフィック・アダプタにも適用されます。

6.8.1 I64 のグラフィック・サポート

V8.2

OpenVMS I64 システム上で現在サポートされているグラフィック・アダプタは、組み込み型 RADEON 7000 PCI アダプタだけです。

ATI RADEON 7500 PCI グラフィック・アダプタは OpenVMS I64 Version 8.2 で近い将来サポートされる予定です。現在テストが進行中です。このグラフィック・カードのサポートが可能になったら、下記の Web サイトで発表されます。

<http://h71000.www7.hp.com/new/>

6.8.2 RADEON 7000 ではハードウェア・アクセラレーション 3D グラフィックはサポートされていない

V8.2

ハードウェア・アクセラレーションによる 3D (OpenGL) レンダリングは、組み込み型 RADEON 7000 PCI アダプタではサポートされていません。

6.9 ATI RADEON 7500 グラフィック

V7.3-2

ここでは、ATI RADEON 7500 グラフィックのリソース要件、拡張機能、修正点および制限事項について説明します。HP DECwindows Motif for OpenVMS に関するドキュメントを参照したい場合、特に『Managing DECwindows Motif for OpenVMS Systems』を参照したい場合は、このドキュメントと、その他のドキュメントを、次の Web サイトから入手できます。

<http://www.hp.com/go/openvms/doc>

OpenVMS Version 8.2 より、3D サポートのライセンス(使用権)は、OpenVMS のライセンスの一部として含まれています。詳細については、第 6.14 節を参照してください。

6.9.1 リソースの要件

RADEON グラフィックのサポートには、次に示す、システム単位のリソースが必要です。

- 2 個のグローバル・セクション
- 296 KB のグローバル・メモリ

さらに、RADEON カードごとに、次のリソースが必要です。

- 3 個のグローバル・セクション
- 16 MB プラス 1 ページのグローバル・メモリ
- 16 MB プラス 1 ページのバッファ・オブジェクト・スペース (32 ビット System Space Windows)

グローバル・セクションの必要量は GBLSECTIONS システム・リソースに加算され、16M バイト強のグローバル・メモリは GBLPAGES リソースと GBLPAGFIL リソースに加算されます。

たとえば、1 枚の RADEON カードには、次のリソースが必要です。

- 5 個のグローバル・セクション
- 16 MB + 8 KB + 296 KB のグローバル・メモリ

これらの数は、次の値に相当します。

GBLSECTIONS	5
GBLPAGES	33376 (GBLPAGES の単位は、512 バイト・ページレット)
GBLPAGFIL	2086 (GBLPAGFIL の単位は、8192 バイトの Alpha ページ)

4 カード構成の場合は、次のリソースが必要です。

- 14 個のグローバル・セクション
- 296 KB + 4*16 MB + 4*8 KB = 64 MB + 328 KB のグローバル・メモリ

これらの数は、次の値に相当します。

GBLSECTIONS	14
GBLPAGES	131728 (GBLPAGES の単位は、512 バイト・ページレット)
GBLPAGFIL	8233 (GBLPAGFIL の単位は、8192 バイト Alpha ページ)

6.9.2 Version 8.2 の新機能

以降の項では、VMS732_GRAPHICS-V0200 ECO キットに含まれていなかった拡張機能について説明します。

6.9.2.1 テキストその他のビットマップの描画性能の改善

V8.2

DMA モードでの、テキストやその他のビットマップの描画操作 (画面深度 16 または 24) は、プラットフォーム、グラフィック・カード、およびバスの種類 (AGP、33 MHz PCI、66 MHz PCI) に依存しますが、以前より 10 ~ 20 倍速くなりました。

6.9.2.2 DMA 操作の性能改善

V8.2

DMA 描画操作の性能は、総じて 4 倍以上に向上しました。この性能向上は、DMA モード (画面深度 16 または 24) にのみ関係します。

6.9.2.3 OpenGL テクスチャ・コピーの性能改善

V8.2

glCopyTexImage2D の性能が大きく改善されました。

6.9.2.4 画面解像度 1024 × 864 の追加

新たに画面の解像度 (ビデオ・モード) として 1024 × 864 が、深度 8、16 および 24 ビット、リフレッシュ・レート 60、70、75、および 85 Hz でサポートされました。

6.9.2.5 ハードウェア・アクセラレーションによるインダイレクト 3D レンダリング

ローカルのトランスポート層経由でクライアント・サイドから使用できる、ハードウェア固有の 3D 描画アクセラレーションであるダイレクト・レンダリングの OpenGL/Mesa 3D ライブラリ (DECW\$MESA3DSHR_RADEON) と同じものが、DECwindows サーバで、ネットワークのトランスポート層を経由して使用できるようになりました。(つまり、DECnet または TCP/IP を使用した、インダイレクト・レンダリングです。)

ハードウェア・アクセラレーションによるインダイレクト・レンダリングを使用する 3D アプリケーションの性能は、ローカルのハードウェア・アクセラレーションによるダイレクト・レンダリングと、ソフトウェアのみによるレンダリング (以前は、リモート・コネクションでは、この方法しかありませんでした) との中間です。

6.9.3 問題の修正 (Alpha のみ)

V8.2

Version 8.2 では、以下の問題が修正されています。

- OpenGL の論理操作 (GL_XOR その他) がラスター描画操作 (glBitmap, glDrawPixels) 用に実装されました。
- 頂点の数が正しくない OpenGL プリミティブ (たとえば、2 頂点しかない GL_TRIANGLE_STRIP、奇数の頂点や 3 つ以下の頂点の GL_QUAD_STRIP、奇数の頂点の GL_LINES) は、接続がローカル (つまり、ダイレクト・レンダリング) であるか、リモートであるかにより、プロセスのクラッシュまたはサーバのクラッシュになることがありました。
- 画面上のウィンドウを移動した後、テキストを再描画するとウィンドウの境界が壊れることがありました。

- DECwindows の起動中に、DRI (ハードウェア・アクセラレーションによるダイレクト・レンダリング) の起動に失敗すると、DECwindows サーバが ACCVIO エラーで終了していました。DECwindows サーバは、DRI の起動に失敗した画面では DRI を無効にして起動するようになりました。
- カーソル幅とカーソル高がゼロで QueryBestSize が呼び出されると、DECwindows サーバは、幅と高さゼロを「ベスト」サイズとして返していました。この影響の 1 つは、従来の DECwindows Session Manager の「Pointer Options」ダイアログ・ボックスで、ポインタの形が表示されませんでした。DECwindows サーバは幅、高さとも 64 で返すようになりました。
- PIO モード (画面深度 8 ビットまたは Open3D ライセンスが無い) では、大量のテキストを出力しているような状況で、サーバがときどきハングすることがありました。
- Radeon DDX が (AGP Radeon ではなく) PCI Radeon カードを検出したときにエラー・ログに出力するメッセージ "vmsInitDevice: Wrong device driver loaded" は、誤解を招くため削除されました。DDX は、明らかに情報通知のみと分かるメッセージ "Radeon: Found PCI device" または "Radeon: Found AGP device" を出力するようになりました。
- DECterm ウィンドウで、テキストをバックスペースで消去すると、消去した文字の一部が画面に残る現象がありました。
- グラフィック・デバイス・ドライバ (SYS\$G*DRIVER) と OPDRIVER の競合のために、ブート時にマシン・チェックが発生することがありました。
この問題は、新しいシステム (DS25, ES45, ES47, ES80, GS80/160/320 /1280) および古いシステムで、コンソールとして使用しているグラフィック・カードが、1 次 PCI バス上にない場合に発生しやすいものでした。

これらの問題はすべて修正されました。

6.9.4 DECW\$OPENGLSHR_RADEON.EXE を DECW\$MESA3DSHR_RADEON.EXE にリネーム

V8.2

共有ライブラリ SYS\$LIBRARY:DECW\$OPENGLSHR_RADEON.EXE は、Mesa 3D コードをベースとしていることを反映させるために、リネームされて SYS\$LIBRARY:DECW\$MESA3DSHR_RADEON.EXE になりました。API と機能は以前のリリースと同じです。新しいファイル指定の共有ライブラリを指定するための論理名 DECW\$OPENGLSHR が定義されています。

6.9.5 サポートの制限事項

V7.3-2

次の機能は、サポートされていません。

- S ビデオ出力
- デジタル出力
- デュアル・ヘッド動作

DVI ポートとアナログ VGA (CRT) ポートの両方にモニタを接続すると、両方のスクリーンに同じビデオ出力が得られます。これを、クローン・ビデオと言います。各ポートに独立したビデオ表示を行う本当の意味でのデュアル・ヘッド動作は、将来のリリースでサポートされます。

6.9.6 高リフレッシュ・レートでの画像への影響

V8.2

高解像度(たとえば、1920 × 1440 や 2048 × 1536)、高リフレッシュ・レート、高負荷の状況では、使用している RADEON カードやモニタによっては、画像に影響が出ることがあります。このような状況が発生したら、リフレッシュ・レートを下げてみてください。

6.9.7 OpenGL は IEEE 演算のみをサポート

V8.2

OpenGL ライブラリは、IEEE 浮動小数点形式のみをサポートしています。VAX 浮動小数点形式はサポートしていません。アプリケーションのコンパイルでは、`/FLOAT=IEEE_FLOAT` オプションを使用してください。

6.9.8 グラフィック・コンソール (OPA0) に出力する際に DECwindows サーバがハングアップ

V8.2

DECwindows サーバの起動後に、グラフィック・コンソール (OPA0) に出力があるると、DECwindows サーバはハングアップし、画面はフリーズします。CTRL/F2 を押すと DECwindows サーバは実行状態に戻ります。

通常、コンソールに表示されるメッセージは、直接 OPA0 に書き込むのではなく、OPCOM と Console Window アプリケーションを使用してメッセージ表示することをお勧めします。このアプリケーションを有効にするには、コマンド・プロシージャ

SYS\$MANAGER:DECW\$PRIVATE_APPS_SETUP.COM を編集し、次のグローバル・シンボル定義を追加します。

```
$ DECW$CONSOLE_SELECTION == "WINDOW"
```

SYS\$MANAGER:DECW\$PRIVATE_APPS_SETUP.COM が存在しない場合は、SYS\$MANAGER:DECW\$PRIVATE_APPS_SETUP.TEMPLATE から作成することができます。

SYS\$MANAGER:DECW\$PRIVATE_APPS_SETUP.COM の編集後、次のコマンドを入力して DECwindows を再起動します。

```
$ @SYS$MANAGER:DECW$STARTUP RESTART
```

6.9.9 モニタは初期化時に接続されていなければならない

RADEON 7500 グラフィック・カードは、ビデオ出力ポートを 2 つ (デジタルとアナログ) 備えています。デジタル・インターフェースは、現在サポートされていません。ただし、デジタル・ツー・アナログ・アダプタを使用すると、アナログ・モニタをデジタル・ポートに接続でき、アナログ・ポートと同じ出力を得ることができます。デジタル・ポートを使用する場合、ポートを正しく動作させるためには、システムの電源投入より前にモニタが接続されていなければなりません。

6.9.10 ブート・リセットの推奨 (Alpha のみ)

コンソール変数 `boot_reset` に ON を設定することをお勧めします。

6.9.11 オーバレイ・プレーンはサポートされない

ハードウェア・オーバレイ・プレーンはサポートされていません。

6.9.12 単一カラーマップのみサポート

RADEON 7500 グラフィック・コントローラがサポートするハードウェア・カラーマップは、1 つだけです。デフォルトのビジュアルが PseudoColor の場合、8 ビット・カラーに変更するときは、このことに留意してください。複数の PseudoColor カラーマップを一度に使用しようとする、カラーマップがフラッシュします。

注意

3D (OpenGL) レンダリングは、8 ビットの PseudoColor ビジュアルではサポートされていません。(第 6.9.18 項も参照してください。)

アプリケーションは、自分でカラーマップのインストールやデインストールを行って
はなりません。これらの動作は、ウィンドウ・マネージャが行います。ただし、ア
プリケーションは、どのカラーマップをインストールまたはデインストールするかにつ
いてのヒントをウィンドウ・マネージャに知らせなければなりません。この情報は、
Xlib 関数の XsetWMColormapWindows() を使用して渡します。この関数は、指定さ
れたウィンドウの WM_COLORMAP_WINDOWS プロパティを設定します。

6.9.13 すべてのウィンドウで同じビット深度

RADEON 7500 カードを使用している場合、特定のヘッド上のすべてのウィンドウ
は、同じビット深度でなければなりません。RADEON 7500 カードは、それぞれの
グラフィック・ヘッドで、8、16、および 24 ビット/ピクセルのビット深度をサポ
ートしています。しかし、DECwindows サーバが特定のヘッド上で一旦ビット深度を
確立すると、そのビット深度のウィンドウやビジュアルだけが作成可能となります。

6.9.14 読み取り/書き込みカラー・マップのピクセル深度

デフォルトでは、RADEON 7500 には、読み取り専用の TrueColor カラー・マップ
が 1 つある 24 プレーンのピクセル深度が用意されています。DECwindows Paint な
どの一部のアプリケーションでは、読み取り/書き込みのカラー・マップが必要です。
Paint は、読み取り専用カラー・マップで実行されると、次のエラー・メッセージを
出力して失敗します。

```
Error: Visual Not Supported  
Supported Visuals are {PseudoColor, GrayScale, StaticGray}
```

読み取り/書き込みのカラー・マップを使用するには
SYS\$MANAGER:DECW\$PRIVATE_SERVER_SETUP.COM ファイル (このフ
ァイルが存在しない場合は、SYS\$MANAGER:DECW\$PRIVATE_SERVER_
SETUP.TEMPLATE から作成します) を編集して、次の論理名定義を追加します。

```
$ DEFINE/EXECUTIVE/SYSTEM/NOLOG DECW$SERVER_PIXEL_DEPTH 8,8,8,8,8,8,8
```

そして、次のコマンドを使用して DECwindows を再起動します。

```
$ @SYS$MANAGER:DECW$STARTUP RESTART
```

この変更により、(マルチ・カード構成が可能な、最大 8 枚のグラフィック・カード
に) ピクセル深度として 8 プレーンが設定され、サーバが PseudoColor ビジュアルを
提供できるようになります。

6.9.15 backing store と save unders は 3D ウィンドウをサポートしていない

RADEON 7500 X11 サーバの backing store と save unders の実装では、3D ウィ
ンドウをサポートしていません。

6.9.16 スレッドの制限事項

OpenVMS 用 RADEON 7500 OpenGL ライブラリは、スレッド・セーフではありません。ただし、OpenGL の使用がプログラム内の単一のスレッドに限定されている場合は、OpenGL をマルチスレッド・プログラムで使用することができます。

6.9.17 シングル・バッファ・ビジュアルがサポートされていない

RADEON 7500 DECwindows サーバは、ダブル・バッファ・ビジュアルだけをサポートしています。シングル・バッファリングを行うには、ダブル・バッファ・ビジュアルを選択して、アプリケーション内で `glDrawBuffer(GL_FRONT)` を呼び出さなければなりません。

6.9.18 カラー・インデックス・モードでの 3D がサポートされていない

DECwindows サーバが 8 ビット深度で起動された場合、8 ビット・ビジュアルが 2D 操作でサポートされますが、OpenGL レンダリングは、8 ビット・ビジュアルではサポートされません。

6.9.19 タイマ・メカニズム

環境によっては、プロセスがハードウェア・ロックを所有している間に、そのプロセスに割り込みが発生することがあります。その結果、DECwindows サーバがハングアップしているように見えることがあります。

このような状況を検出し、一時停止中のプロセス内でイメージを強制的に一時停止から抜け出させるか、場合によってはプロセスを削除してこのような状況から回復させるために、タイマ・メカニズムが実装されました。

タイマ・メカニズムは、次の 2 つの論理名を使用して構成できます。これらの論理名は、`DECW$PRIVATE_SERVER_SETUP.COM` ファイルで指定しなければなりません。

- `DECW$DRM_TIMER_PERIOD` (デフォルト: 5.0 秒)
クロック・ティックの間隔を秒数で指定します。浮動小数点数です。
- `DECW$DRM_TIMEOUT` (デフォルト: 6)
タイムアウトが発生して、DECwindows サーバが RADEON カードの制御を奪うまでに待つクロック・ティック数を指定します。

デフォルト値は、グラフィック・アプリケーションの性能にタイマが与える影響を最小限にするように選択されています。DECwindows サーバが応答を再開するまでの時間を短くしたい場合は、`DECW$DRM_TIMER_PERIOD` の値を小さくします。

次のいずれかの状況では、ハードウェア・ロックの保持中に、プロセスに割り込みが発生することがあります。

- プロセスは、別システム上に表示された端末を使用して、リモートでログインされている。
- プロセスは、通常の終了処理が実行されない状態で、他のプロセスによって一時停止または終了させられたサブプロセスである。

どちらの状況も発生しないと思われる構成では、時間間隔にゼロを設定して、タイマ・メカニズムを完全に無効にすることができます。

```
$ DEFINE/SYSTEM DECW$DRM_TIMER_PERIOD 0
```

ECW\$DRM_TIMER_PERIOD の値を変更するたびに、DECwindows サーバを再起動するか、システムをリブートして、変更内容を有効にしなければなりません。DECwindows サーバを再起動するには、次のコマンドを使用します。

```
$ @SYS$STARTUP:DECW$STARTUP RESTART
```

6.10 DECwindows X11 ディスプレイ・サーバ (Alpha のみ)

ここでは、OpenVMS Alpha システムに対応した DECwindows X11 ディスプレイ・サーバに関する注意事項について説明します。

6.10.1 社名とオペレーティング・システム・バージョン情報の更新

V8.2

社名とオペレーティング・システムのバージョンに関して、多数の箇所で更新しています。

6.10.2 S3 マルチヘッド・グラフィック

永続的な制限事項

S3 Trio32 または Trio64 グラフィック・カードを装備している Alpha コンピュータでは、シングル・スクリーン・ディスプレイだけがサポートされます。マルチヘッド・グラフィックはサポートされません。

6.10.3 グラフィック・ボードのサポート

V7.3-2

OpenVMS Version 7.3 から、OpenVMS オペレーティング・システムに新たにグラフィック・ボード・サポートが統合されています。サポートされる新しいボードは、次のとおりです。

- ATI RADEON 7500 (2003 年に追加)
- Elsa GLoria (PowerStorm 4D10T)
- OXYGEN VX1
- PowerStorm 300
- PowerStorm 350

6.11 DIGITAL Modular Computing Components (DMCC)

ここでは、DMCC に関する注意事項について説明します。

6.11.1 Alpha 5/366 および 5/433 PICMG SBC の制限事項

永続的な制限事項

KZPDA SCSI コントローラおよび PBXGA グラフィック・カードは、DIGITAL Modular Computing Components (DMCC) Alpha 5/366 および 5/433 PICMG SBC のブリッジの後ろにあるスロットに挿入することはできません。

6.11.2 SRM コンソールの更新

永続的な制限事項

Alpha 4/233 (21064a)、4/266 (21164a)、5/366、5/433 DMCC システムで SRM コンソールを更新するには、SRM コンソールまたは AlphaBIOS セットアップを選択しなければなりません。格納できるコンソールは 1 つだけです。

- これらのシステムで OpenVMS を実行している場合は、SRM コンソールだけを更新する。
- これらのシステムで Windows NT を実行している場合は、AlphaBIOS セットアップだけを更新する。

SRM と AlphaBIOS コンソールの両方を誤って更新すると、AlphaBIOS セットアップ・メニューが表示されます。この後、SRM コンソールに戻るオプションはありません。AlphaBIOS セットアップ・メニューを終了し、SRM コンソールに戻るに

は、次のインターネット・サイトにある Firmware Update ユーティリティを使用しなければなりません。

<ftp://ftp.digital.com/pub/Digital/Alpha/firmware/index.html>

6.12 Digital Personal Workstation: OpenVMS V7.3-1 およびそれ以降のブート

V7.3-1

Digital Personal Workstation 433au, 500au, および 600au シリーズのシステムを使用するときは、コントローラ・チップが Cypress PCI Peripheral Controller の場合、IDE CD-ROM から OpenVMS Version 7.3-1 以降をブートできます。Intel Saturn I/O (SIO) 82378 チップが実装されている Digital Personal Workstation au シリーズ・システムでは、IDE CD-ROM から OpenVMS をブートすることはできません。Intel SIO チップが実装されている場合は、SCSI CD-ROM を使用しなければなりません。

どちらの IDE チップが実装されているか調べるには、次の SRM コンソール・コマンドを入力します。

```
SHOW CONFIGURATION
```

「Cypress PCI Peripheral Controller」と表示された場合は、OpenVMS をブートできます。

「Intel SIO 82378」と表示された場合は、SCSI CD-ROM を使用して CD-ROM からブートする必要があります。

6.13 I/O 負荷が重い場合、LUN が多数あるデュアル・コントローラ HSGnn に障害が発生することがある

V7.3-2

ドライバの性能が改善されたことと、システムの高速化によって、比較的多数の LUN が構成されているデュアル・コントローラ HSGnn ストレージ・アレイで扱うことができる I/O 量の制限が取り払われました。この制限値に達した場合、アレイは、コントローラ間の通常の定期的な同期処理を完了できないほど、I/O 処理がビジーになることがあります。これにより、コントローラのハングアップまたは障害が発生し、手作業でコントローラをリセットするまで、一部またはすべての LUN がホストからアクセスできなくなることがあります。このようなコントローラ障害が発生したときには、HSG に報告される Last Failure Code は、多くの場合 01960186, 01942088, および 018600A0 になります。

大半の HSGnn デバイスは、問題なく動作し続けます。負荷が重く、約 24 個以上の LUN が HSG に構成されていて、HSG コントローラのハングアップや障害が発生するサイトでは、コントローラを再構成して、LUN の数を少なくするか、HSG の負荷が重くならないように I/O を分散させると、ハングアップや障害を防止できる可能性があります。

この問題は、弊社のエンジニア・グループで調査を行っています。

6.14 Open3D グラフィックのライセンス方式の変更

V8.2

これまで、3D グラフィック表示機能は、OpenVMS オペレーティング・システムとは別にライセンスしていました。Open3D レイヤード・プロダクトは、最初の提供時から、オペレーティング・システムとは別のライセンスが必要でした。Open3D ソフトウェアを OpenVMS オペレーティング・システムの一部として出荷を開始したときも、3D グラフィック表示機能は、別のライセンス機能とされました。このようなライセンス方式の例としては、3X-PBXGG-AA ATI RADEON 7500 PCI 2D/3D グラフィック・アダプタによるグラフィック表示機能をサポートするための Open3D ライセンスがあります。

OpenVMS Version 8.2 からは、AlphaServers と Integrity サーバのいずれについても、3D グラフィック表示機能はオペレーティング・システムとともにライセンスされます。したがって、OpenVMS Version 8.2 には Open3D の個別のライセンスはありません。以前のバージョンの OpenVMS で 3D 表示操作を行うためには、Open3D ライセンスをシステムにインストールする必要があります。

弊社は、標準契約または Mature Product Support の OpenVMS Version 7.3-2 とともに出荷された 3D デバイス・ドライバについては、サポート契約内容に従ってサポートを継続します。以下のアダプタのデバイス・ドライバは、OpenVMS Version 7.3-2 とともに出荷されています。

- PowerStorm 300 および 350 PCI グラフィック・アダプタ (SN-PBXGD)
- ATI RADEON 7500 PCI および AGP グラフィック・アダプタ (3X-PBXGG)

これらのアダプタは OpenVMS Version 8.2 でも 3D グラフィック表示が可能ですが、ライセンスは不要です。さらに、OpenVMS Version 8.2 では、以下の 2D グラフィック・アダプタもサポートを継続します。

- ELSA Gloria Synergy (SN-PBXGK)
- 3Dlabs Oxygen VX1 (SN-PBXGF)

ATI RADEON 7500 PCI グラフィック・アダプタは、OpenVMS I64 Version 8.2 でも近日中にサポートされる予定です。現在はテストを実施中です。このグラフィック・カードがサポートされるようになった時点で、次の Web サイトでお知らせします。

<http://h71000.www7.hp.com/new/>

6.15 OpenVMS 用の PowerStorm 300/350 PCI グラフィック・サポート

V8.2

OpenVMS Alpha を実行している Compaq Workstation での PowerStorm 300 /350 PCI グラフィック・コントローラのサポートに関する注意事項については、『PowerStorm 300/350 OpenVMS Graphics Release Notes Version 2.0』を参照してください。リリース・ノート、およびインストール・ガイドは、グラフィック・カードとともに発送されます。

Open3D のライセンスはチェックされない

OpenVMS Version 8.2 以降では、3D (OpenGL) サポートのライセンスは、OpenVMS のライセンスの一部として含まれています。詳細は、第 6.14 節を参照してください。

論理名 DECW\$OPENGL_PROTOCOL の定義

3D グラフィック・アプリケーションを実行して、PowerStorm 350/300 グラフィック・カードを備えたシステムに出力を表示する場合、アプリケーションを実行しているシステム上で、論理名 DECW\$OPENGL_PROTOCOL が次のように定義されていることを確認してください。

```
$ DEFINE DECW$OPENGL_PROTOCOL DECW$OPENGL_PROTOCOL_V11
```

問題の修正

P350 はセッションの終了時に、再初期化に失敗することがありました。

この問題は以下の 2 点の変更で修正されました。

- デバイス固有の riCloseScreen 関数に vmsCloseScreen の呼び出しを追加しました。riCloseScreen 関数は、CDE セッションの終了時などに呼び出されます。vmsCloseScreen は GB デバイスへのチャンネルの割り当てを解除し、ドライバがボードを正しく再初期化できるようにします。
- デバイス・ドライバのピクセル・コンバータ同期アルゴリズムが大きく改善されました。

6.16 RFnn DSSI ディスク・デバイスとコントローラ・メモリのエラー

V6.2

RF31T, RF31T+, RF35, RF35+, RF73, RF74 DSSI ディスク・デバイスの以前のバージョンのマイクロコードには問題があります。この問題が原因で、データが失われる可能性があり、これらのデバイスからデータを読み込むときに、デバイスでコントローラ・メモリ・エラー (エラー検出/訂正 (EDC) エラーとも呼ばれる) が発生していた場合、問題が発生することがあります。このエラーは仮想サーキットの閉鎖やハードウェアの障害が原因で発生している可能性があります。

これらのデバイスを使用する場合は、マイクロコードのリビジョン・レベルを確認してください。マイクロコードのリビジョン・レベルが表 6-2 に示されている値より低い場合は、マイクロコードを更新してください。

RF31T, RF31T+, RF35+ 以外のすべてのモデルのマイクロコードは、最新の OpenVMS バイナリ・ディストリビューション CD-ROM にあります。

DSSI ディスク・デバイスのマイクロコード・リビジョン・レベルを表示するユーティリティ・プログラムである RF_VERS ユーティリティも同じ CD-ROM に格納されています。このユーティリティ・プログラムの使用法とマイクロコードの更新方法については、ここで説明します。

注意

RF31T, RF31T+, RF35+ ディスク・ドライブを使用し、マイクロコードのバージョンがサポートされないバージョンであり (表 6-2 を参照), サポート契約を結んでいるお客様の場合には、弊社のサポート担当者にお問い合わせください。サポート契約を結んでいないお客様の場合には、公認代理店にお問い合わせください。

DSSI ディスクのマイクロコードのうち、表 6-2 に示したリビジョン・レベル以上がサポートされます。

表 6-2 サポートされるマイクロコードのレベル

デバイス・タイプ	サポートされるマイクロコードの最小リビジョン・レベル
RF31T	T387E
RF31T+	T387E
RF35	T392D
RF35+	T392D
RF36	V427P
RF73	T392D
RF74	V427P

DSSI ディスク・デバイスのマイクロコード・リビジョン・レベルを表示するには、次の手順を実行します。

1. SYSTEM アカウントにログインするか、または CMKRNL 特権、DIAGNOSE 特権、SYSPRV 特権がある他のアカウントにログインします。
2. 次のコマンドを入力します。

```
$ SET PROCESS /PRIVILEGE=(DIAGNOSE,CMKRNL,SYSPRV)
$ SHOW DEVICE FYA0:
```

VAX システムで SHOW DEVICE コマンドを実行してエラーが発生した場合には、次のコマンドを入力します。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONN FYA0/NOADAP
SYSGEN> ^Z
```

Alpha システムで SHOW DEVICE コマンドを実行してエラーが発生した場合には、次のコマンドを入力します。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> IO CONNECT FYA0: /NOADAP
SYSGEN> ^Z
```

次の例に RF_VERS ユーティリティが出力する内容を示します。

```
Program Name:  RF_VERS
Revision Level: V1.2s

NOTICE: This program does not currently support the RF72 or any
        HSDxx controllers. See next version for support.

DSSI disks currently on this system as seen by RF_VERS

Device      Node      Status   Hardware  Firmware
Name        Name
-----
_$22$DIA7:  R4JL2I   mounted  RF73      T387A
_$22$DIA6:  R4I0BG   mounted  RF73      T387A
_$22$DIA8:  R4XLWE   mounted  RF73      T387A
_$22$DIA2:  R4FCZK   mounted  RF73      T387A
_$22$DIA3:  R4CKCG   mounted  RF73      T387A
_$22$DIA4:  R4ZKUE   mounted  RF73      T387A
_$22$DIA9:  R4GYI    mounted  RF73      T387A
_$22$DIA1:  R4XRYI   mounted  RF73      T387A
```

デバイスのマイクロコードを更新するには、表 6-3 でデバイスとプラットフォームに対応するコマンドを確認し、使用します。

重要

マイクロコードを更新する場合は、あらかじめディスクのバックアップを作成してください。

表 6-3 DSSI ディスク・デバイスのマイクロコードを更新するコマンド

デバイス・タイプ	プラットフォーム	コマンド
RF35	Alpha	\$RUN SYS\$ETC:RF35_T392F_DEC_ALPHA.EXE
RF35	VAX	\$RUN SYS\$ETC:RF35_T392F_DEC.EXE
RF36	Alpha	\$RUN SYS\$ETC:RF36_V427P_DEC_ALPHA.EXE
RF36	VAX	\$RUN SYS\$ETC:RF36_V427P_DEC.EXE
RF73	Alpha	\$RUN SYS\$ETC:RF73_T392F_DEC_ALPHA.EXE
RF73	VAX	\$RUN SYS\$ETC:RF73_T392F_DEC.EXE
RF74	Alpha	\$RUN SYS\$ETC:RF74_V427P_DEC_ALPHA.EXE
RF74	VAX	\$RUN SYS\$ETC:RF74_V427P_DEC.EXE

重要

SCSI_INFO.EXE, RF_VERS.EXE をはじめ、表 6-3 に示されているファイルは絶対に削除しないでください。これらのファイルを削除すると、VAX システムでは VMSKITBLD.COM がファイルを検索することができなくなります。同様に、Alpha システムでは AXPVMS\$PCSI_INSTALL と AXPVMS\$PCSI_INSTALL_MIN での PRODUCT INSTALL コマンドが失敗します。

6.17 RZnn ディスク・ドライブの考慮事項

ここでは、各種の RZ ディスク・ドライブに関する注意事項について説明します。

6.17.1 RZ25M と RZ26N ディスク・ドライブ: 推奨事項

V7.1

DWZZA とロング・ディファレンシャル SCSI バスを含む構成を使用して、弊社がサポートする SCSI ディスク・ドライブをテストしたところ、2 台のドライブ (RZ25M と RZ26N) でバス・フェーズに関する問題が検出されました。そのため、DWZZA を接続するディファレンシャル・バスの長さが 20 メートル以上の構成では、これらのドライブを使用しないでください。

この勧告は RZ25M および RZ26N ドライブにのみ適用されます。OpenVMS SPD に、サポートされるドライブとしてリストされている他のすべてのディスク・ドライブは、SCSI-2 仕様の上限のバスの長さまで使用できます。

6.17.2 RZ26N および RZ28M ディスク: 推奨ファームウェア・サポート

V6.2-1H3

RZ26N および RZ28M ディスクを使用する場合には、ファームウェアのリビジョン・レベルは 0568 以上をお勧めします。

これらのディスクで最新のファームウェア・リビジョン・レベルが使用されていない場合には、問題が発生する可能性があります。

6.17.3 RZ26L および RZ28 ディスク: マルチホストで使用するために必要なファームウェア

V6.2

OpenVMS Cluster のマルチホスト SCSI バスに RZ26L または RZ28 ディスクを取り付ける場合、ディスクに必要なファームウェア・リビジョンは、最低でも 442 です。

ここでは、一部の RZ26L および RZ28 ドライブでファームウェアを更新するために使用する手順について説明します。この手順を使用できるのは、ドライブがホスト・システムの SCSI アダプタに直接接続されている場合だけです。インテリジェント・コントローラ (HSZ40 や KZPSC など) を介して接続されているドライブは、この手順では更新できません。ファームウェアの別の更新手順があるかどうかについては、インテリジェント・コントローラのドキュメントを参照してください。

重要

ファームウェア・リビジョン・レベル 442 に安全にアップグレードできるのは、特定の RZ26L および RZ28 ファームウェア・リビジョンだけです。使用中のディスクをファームウェア・リビジョン・レベル 442 にアップグレードできるかどうか判断するには、第 6.17.3.1 項を参照してください。ディスクがファームウェア・リビジョン・レベル 442 をサポートできる場合は、第 6.17.3.2 項で説明している RZTOOLS ユーティリティを使用して、ディスクのファームウェアを更新します。

6.17.3.1 ファームウェア・リビジョン・レベル 442 の必要条件

ファームウェア・リビジョン・レベル 442 に安全にアップグレードできるのは、表 6-4 に示したディスク・ドライブとファームウェア・リビジョン・レベルの組み合わせだけです。他の組み合わせで更新手順を実行すると、ディスクを永久に破損する可能性があります。

表 6-4 リビジョン・レベル 442 ファームウェアの互換性

ディスク・ドライブ	ファームウェア・リビジョン	ディスク・ファイル名
RZ26L	440C	RZ26L_442D_DEC.FUP
RZ28	441C または D41C 435 または 436	RZ28_442D_DEC2104.FUP RZ28P4_442C_DEC.FUP

6.17.3.2 ファームウェア・リビジョン・レベル 442 のインストール手順

使用しているディスクでリビジョン・レベル 442 ファームウェアが必要かどうか、安全にアップグレードできるかどうかを判断した後、次の手順を実行してファームウェアを更新します (アップグレードするディスクのファイル名については、表 6-4 を参照してください)。

```
$ RZTOOLS_ALPHA := $SYS$ETC:RZTOOLS_ALPHA
$ RZTOOLS_ALPHA DKB500 /LOAD=SYS$ETC:filename.FUP
Read in 262144 bytes.
Current FW version - X440C
Upgrading to      - DEC0
Loading code .....
New code has been sent to the drive.
```

6.18 ZLX グラフィック・ボードのサポート

V8.2

次のグラフィック・コントローラ・ボード・ファミリーは、OpenVMS Version 8.2 ではサポートされていません。

- ZLX-M シリーズ (PixelVision): ZLX-M1 (PMAGC-AA), ZLX-M2 (PMAGC-BA)
- ZLX-L シリーズ (PixelVision Lite): ZLX-L1 (PMAGC-DA), ZLX-L2 (PMAGC-EA)
- ZLXp-L シリーズ (PixelVision PCI): ZLXp-L1 (PBXGC-A), ZLXp-L2 (PBXGC-B)

OpenVMS Version 8.2 では、以下のグラフィック・コントローラ・ボード・ファミリー用に、OpenVMS と一緒に出荷される基本的な 2D 機能を使用した 2D サポートのみが提供されます。以下のボード用に 2D サポートを利用する目的で Open3D をインストールしないでください。

- ZLX-E シリーズ (FFB): ZLX-E1 (PMAGD-AA), ZLX-E2 (PMAGD-BA), ZLX-E3 (PMAGD-CA)
- ZLXp-E シリーズ (TGA): ZLXp-E1 (PBXGA-A), ZLXp-E2 (PBXGA-B), ZLXp-E3 (PBXGA-C)
- ZLX2-E シリーズ (TGA2): PowerStorm 3D30 (PBXGB-AA), PowerStorm 4D20 (PBXGB-CA)

第 A.1 節の関連する注意事項も参照してください。

6.19 OpenVMS デバイス・ドライバの再コンパイルと再リンク

ここでは、OpenVMS デバイス・ドライバの再コンパイルと再リンクに関する注意事項について説明します。

第 5.3 節の関連する注意事項も参照してください。

6.19.1 Alpha および VAX の SCSI デバイス・ドライバ

V7.3-1

OpenVMS の以前のバージョンのすべての OpenVMS Alpha SCSI デバイス・ドライバが OpenVMS Version 7.3-1 以降で正しく動作するには、再コンパイルと再リンクが必要です。

OpenVMS Alpha Version 7.0 より前のバージョンからアップグレードしている OpenVMS Alpha SCSI ドライバがある場合は、第 6.19.2 項を参照してください。

OpenVMS Version 7.1 では、すべての OpenVMS VAX SCSI デバイス・ドライバの再コンパイルと再リンクが必要でした。OpenVMS Version 7.1 で実行できるように再コンパイルと再リンクされた OpenVMS VAX デバイス・ドライバは、OpenVMS Version 7.3 以降でも正しく動作します。

6.19.2 OpenVMS Alpha デバイス・ドライバ

V7.1

OpenVMS Alpha Version 7.0 で実行できるように再コンパイルおよび再リンクされたデバイス・ドライバは、OpenVMS Alpha Version 7.1 以降で実行できるようにするためにソース・コードを変更したり、再コンパイルや再リンクしたりする必要があります(ただし、Alpha SCSI ドライバについては、再コンパイルと再リンクが必要です。第 6.19.1 項を参照してください)。

OpenVMS Alpha Version 7.0 より前のリリースのデバイス・ドライバのうち、OpenVMS Alpha Version 7.0 に対応するよう再コンパイルおよび再リンクされていないデバイス・ドライバを OpenVMS Alpha Version 7.1 以降で実行するには、再コンパイルと再リンクが必要です。

OpenVMS Alpha Version 7.0 では、OpenVMS Alpha 特権インタフェースと構造体が大幅に変更されました。これらの変更の結果、OpenVMS Alpha Version 7.0 より前のリリースのデバイス・ドライバでは、OpenVMS Alpha Version 7.0 以降で正しく動作するように、ソース・コードを変更する必要があります。カスタマが作成した

ドライバのソースの変更が必要となる OpenVMS Alpha Version 7.0 の変更点の詳細については、『OpenVMS Guide to Upgrading Privileged-Code Applications』を参照してください。

6.20 MON バージョンのデバイス・ドライバの処理

V7.3

OpenVMS Version 7.3 では、SYSTEM_CHECK を有効化すると、SYS\$nnDRIVER_MON.EXE という形式の名前を持つデバイス・ドライバ・イメージがシステム・ローダによって自動的に読み込まれます。対応する_MON バージョンが存在しない場合は、デフォルトのイメージ名 SYS\$nnDRIVER.EXE が使用されます。

6.21 スレッド単位のセキュリティが Alpha デバイス・ドライバに与える影響

V7.2

スレッド単位のセキュリティが OpenVMS Alpha デバイス・ドライバに与える影響については、第 5.3.7 項を参照してください。

6.22 OpenVMS Alpha ドライバのデバイス IPL の設定

V6.2

PCI, EISA, ISA バスをサポートする Alpha ハードウェア・プラットフォームでは、20 または 21 という異なる IPL で I/O デバイスへの割り込みが発生します。デバイスへの割り込みが発生する IPL は、デバイスをプラットフォーム間で移動したときに変わる可能性があります。ドライバがデバイス IPL を 20 であると宣言した後、I/O デバイスへの割り込みが IPL 21 で発生するマシンでそのドライバを実行すると、問題が発生します。

この問題に対する最も簡単な対処法は、PCI, EISA, ISA のデバイス・ドライバで IPL 21 を使用することです。この方法は、I/O デバイスへの割り込みが IPL 20 で発生するプラットフォームでも、I/O デバイスへの割り込みが IPL 21 で発生するプラットフォームでも、正しく動作します。

OpenVMS Alpha の将来のリリースでは、ドライバがデバイス IPL を動的に判断するための、プラットフォームに依存しない機能が提供される予定です。

6.23 CRCTX ルーチンの機能の強化

V7.1-2

Counted Resource Context Block (CRCTX) 構造体の管理に使用できるシステム・ルーチンが強化されました。次のルーチンが、CRCTX 構造体のステータス (CRCTX\$V_ITEM_VALID) を設定およびチェックするようになりました。

- IOC\$DEALLOC_CRCTX
- IOC\$ALLOC_CNT_RES
- IOC\$DEALLOC_CNT_RES
- IOC\$LOAD_MAP

これらのルーチンは次のように変更されました。

有効な CRCTX ステータス (CRCTX\$V_ITEM_VALID を 1 に設定) で IOC\$DEALLOC_CRCTX を呼び出すと、サービスは不正なステータスを返します。SYSBOOT パラメータ SYSTEM_CHECK が設定されている場合には、システム障害が発生します。このため、割り当てが解除されていない有効なリソースがあるときに、ユーザが CRCTX の割り当てを誤って解除することを防止できます。

IOC\$ALLOC_CNT_RES は、無効な CRCTX ステータス (CRCTX\$V_ITEM_VALID を 0 に設定) で呼び出さなければなりません。有効なステータスでこのルーチンを呼び出すと、OpenVMS はこの CRCTX によってマップされたリソースをユーザが手放すものと解釈します。OpenVMS は新しいリソースを割り当てず、不正なステータスを返します。SYSTEM_CHECK が設定されている場合には、システム障害が発生します。IOC\$ALLOC_CNT_RES は有効ビットを設定してから戻ります。

IOC\$DEALLOC_CNT_RES は、有効な CRCTX ステータス (CRCTX\$V_ITEM_VALID を 1 に設定) で呼び出さなければなりません。無効な CRCTX で IOC\$DEALLOC_CNT_RES を呼び出すと、OpenVMS は他のパラメータが有効でないものと解釈し、不正ステータスを返します。SYSTEM_CHECK がセットされている場合には、システム障害が発生します。IOC\$DEALLOC_CNT_RES は有効ビットをクリアしてから戻ります。

IOC\$LOAD_MAP は有効な CRCTX で呼び出さなければなりません。無効な CRCTX (CRCTX\$V_ITEM_VALID を 0 に設定) で呼び出すと、他のパラメータも無効であると解釈され、不正ステータスが返されます。SYSBOOT パラメータ SYSTEM_CHECK がセットされている場合には、システム障害が発生します。

これらの変更により、デバイス・サポート・アプリケーションや特権付きコード・アプリケーションの開発者は、OpenVMS で汎用リソースとして取り扱われる scatter gather レジスタの割り当てを解除する必要があるかどうか判断できます。CRCTX\$V_ITEM_VALID ビットがセットされている場合は、IOC\$DEALLOC_CNT_RES を呼び出さなければなりません。

リタイア製品情報

この付録では、すでにサポートが中止された OpenVMS 製品およびサポートの中止が予定されている OpenVMS 製品についてお知らせします。また、アーカイブ扱いになったマニュアルを検索する方法についても説明します。

フリーウェア

製品のサポートが中止されると、弊社はそれらの製品に対する問題に関する報告を受け付けず、またそのような報告への対処も行いません。しかし、独自の開発やサポートを目的に、以前の製品のソース・コードを必要とされるお客様に対しては、契約上その他の状況で受け入れ可能な場合、多くの旧製品のインストール・キットやソース・コードを OpenVMS Freeware として次の形態で提供しています。

- OpenVMS オペレーティング・システムに添付される Freeware の媒体
- 次のアドレスの Web サイト

<http://www.hp.com/go/openvms/freeware/>

A.1 Compaq Open3D レイヤード・プロダクトは Version 8.2 ではサポートされない

V8.2

OpenVMS Version 8.2 では、レイヤード・プロダクトの Compaq Open3D for OpenVMS Alpha V4.9B およびそれ以前のバージョンは、サポートされなくなりました。ただし、Mature Product Support (MPS) の下で OpenVMS Versions 6.2 および 7.3-2 でのサポートを継続します。この製品と、OpenVMS Version 8.2 に統合され、PowerStorm 300, PowerStorm 350, および ATI RADEON などの新しいカードをサポートする Open3D 製品とを混同しないようにしてください。サポートされなくなった製品は、OpenVMS Version 7.3-2 以降のリリースでの PixelVision, FFB, TGA, および TGA2 ベースのグラフィックの 3D サポートです。第 6.18 節の関連する注意事項を参照してください。

サポートされていない Open3D 製品がシステムにインストールされている場合は、問題が発生する可能性があります。たとえば、DECwindows ディスプレイ・サーバは、CPU を多用するループ内でハングアップすることがあります。使用中のバージョンの OpenVMS に Open3D がすでにインストールされている場合に、OpenVMS Version 8.2 へアップグレードする予定であれば、ターゲット・システムに次のグラフィック・コントローラ・ボードがインストールされていないことをまず確認しなければなりません。

- ZLX-M シリーズ (PixelVision): ZLX-M1 (PMAGC-AA) , ZLX-M2 (PMAGC-BA)
- ZLX-L シリーズ (PixelVision Lite): ZLX-L1 (PMAGC-DA) , ZLX-L2 (PMAGC-EA)
- ZLXp-L シリーズ (PixelVision PCI): ZLXp-L1 (PBXGC-A) , ZLXp-L2 (PBXGC-B)

A.2 Open3D グラフィックのライセンス方式の変更

V8.2

OpenVMS Version 8.2 から , 3D グラフィック表示機能のライセンスは , AlphaServer と Integrity サーバのいずれの場合も , オペレーティング・システムのライセンスに含まれます。したがって , OpenVMS Version 8.2 では Open3D の単独のライセンスは入手できません。

詳細については第 6.14 節を参照してください。

A.3 DECams は OpenVMS Version 8.2 ではサポートされない

V8.2

OpenVMS Version 8.2 では , DECams はサポートされません。弊社では , DECams の代わりに Availability Manager を使用することをお勧めします。Availability Manager についての情報は , 次の Web サイトを参照してください。

<http://h71000.www7.hp.com/openvms/products/availman/>

A.4 DECevent はサポートされない

V8.2

DECevent と DIAGNOSE コマンドは , OpenVMS Version 8.2 ではサポートされません。

DECevent の代替機能としては , 次のツールがあります。

- System Event Analyzer (SEA)

SEA は , OpenVMS とそれ以降のハードウェア・プラットフォーム (たとえば , すべての I64 プラットフォーム , および Alpha の DSnn , ESnn , および大半の GSnn プラットフォーム) でサポートされるエラー・ログ分析ツールになりました。

SEA を含め、最新のハードウェア・サポートを利用するには、入手できる最新の WEBES をインストールしてください。最低でも、Alpha システムでは WEBES Version 4.2 以上、I64 システムでは WEBES Version 4.4 以上が必要です。

SEA のオペレーティング・システム要件と、サポートされるハードウェアについての詳細は、WEBES のインストレーション・ガイドを参照してください。このドキュメントは、他の WEBES ドキュメントとともに、次の Web サイトにあります。

<http://www.hp.com/services/webes>

- Error Log Viewer (ELV)

ELV を使用すると、新しいハードウェア・プラットフォーム上で作成されたエラー・ログ・ファイルを素早く調べることができます。ELV は OpenVMS オペレーティング・システムに統合されており、DCL コマンドの ANALYZE/ERROR_LOG/ELV を入力することでアクセスできます。

ELV 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 ELV の章を参照してください。

A.5 2004 年 12 月に DECwrite のサービス期間が終了

V8.2

DECwrite は、2004 年 12 月 31 日でサービス期間が終了し、Software Product Library から削除されました。

A.6 Error Log Report Formatter (ERF) のサポート終了

V8.2

Error Log Report Formatter (ERF) はサポートされなくなりました。OpenVMS Version 7.2 より前のシステムで作成されたログを、この製品で処理する必要がある場合は、次の Freeware Web サイトから ERF ドキュメントにアクセスできます。

<http://www.hp.com/go/openvms/freeware/>

ERF は Error Log Viewer (ELV) に置き換わりました。詳細については、オンラインヘルプで ANALYZE/ERROR_LOG/ELV を参照するか、『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。

ERF を使用する前に、ELV または DECevent の一部である Binary Error Log Translation ユーティリティを使用してエラー・ログ・ファイルを変換する必要があります。DECevent と DIAGNOSE コマンドはサポートされなくなりましたが、これらのツールを必要とするユーザは、次の Freeware Web サイトから、ソフトウェアと関連ドキュメントをダウンロードすることができます。

<http://www.hp.com/go/openvms/freeware/>

DECevent ソフトウェアをインストールするまでは、DIAGNOSE コマンドを使用するとエラーが出力されます。

A.7 ISA_CONFIG.DAT のサポートの将来のリリースでの中止

V7.1

SYS\$MANAGER:ISA_CONFIG.DAT ファイルを使用して ISA デバイスを構成する機能は、OpenVMS Alpha の将来のリリースではサポートされなくなります。このファイルを使用している場合は、コンソールから ISACFG ユーティリティと、デバイス・ドライバをロードするための新しいファイル・ベースの自動構成方式を使用するように変換しなければなりません (『Writing OpenVMS Alpha Device Drivers in C』を参照)。

A.8 POSIX 1003.4a Draft 4 インタフェースのサポート中止

V7.0

Compaq POSIX Threads Library (以前の DECthreads) の POSIX 1003.4a, Draft 4 ("d4") インタフェースは、将来のリリースでサポートされなくなる予定です。POSIX 1003.4a, Draft 4 インタフェースを使用して作成されたアプリケーションは、POSIX スレッド・ライブラリで提供される新しい POSIX 1003.1c 標準 ("pthread") インタフェースに移行する必要があります。このリリースでは、移行を支援するために、Draft 4 POSIX 1003.4a インタフェース用の互換モードが提供されます。この互換モードは、将来のリリースでは削除されます。

A.9 ビジュアル・スレッド: Version 8.2 ではサポートされない

V8.2

OpenVMS I64 システムや OpenVMS Alpha Version 8.2 ではビジュアル・スレッドはサポートされません。OpenVMS Alpha Version 7.3-2 が、ビジュアル・スレッドがサポートされる最後のバージョンです。

A.10 アーカイブ扱いのマニュアル

V8.2

製品の製造が中止され、オペレーティング・システムが進化するにつれて、OpenVMS マニュアルの一部はアーカイブ扱いになります。アーカイブ扱いのマニュアルはそれ以降保守されず、OpenVMS ドキュメンテーション・セットには含まれません。ただし、次の Web サイトに保管されています。

<http://www.hp.com/go/openvms/doc>

左のサイド・バーにある“Archived documents”をクリックすると、アーカイブ扱いのマニュアルを参照できます。

OpenVMS Version 8.2 のリリースに伴って、以下のマニュアルがアーカイブ扱いになりました。

- 『VAX MACRO and Instruction Set Reference Manual』
- 『OpenVMS VAX RTL Mathematics (MTH\$) Manual』
- 『OpenVMS VAX System Dump Analyzer Utility Manual』

インターロックされたメモリ命令の使用 (Alpha のみ)

インターロックされたメモリ命令を使用するための厳密な規則は、『Alpha Architecture Reference Manual, Third Edition (AARM)』に説明されています。Alpha 21264 (EV6) プロセッサと将来のすべての Alpha プロセッサは、これらの規則で決められている必要条件に関して、以前のプロセッサよりさらに厳しくなっています。この結果、以前は規則に準拠していなくても正常に動作していたコードが、21264 以上のプロセッサを搭載したシステムでは正しく実行できないことがあります。このような規則に準拠していないコード・シーケンスが発生することは、非常にまれであると考えられています。21264 プロセッサは、OpenVMS Alpha Version 7.1-2 より前のバージョンではサポートされません。

規則に従っていないコードを実行すると、インタープロセッサ・ロックが使用される際に、プロセッサ間の同期が失われる可能性があり、インターロックされたシーケンスが常にエラーになる場合は、無限ループになることがあります。BLISS コンパイラの以前のバージョン、MACRO-32 コンパイラと MACRO-64 アセンブラの一部のバージョンでコンパイルされたプログラムや、一部の HP C および C++ プログラムのコード・シーケンスで、このような動作が発生することがあります。

影響を受けるコード・シーケンスでは、LDx_L/STx_C 命令を、アセンブリ言語ソースで直接、またはコンパイラで生成されたコードで使用しています。インターロックされた命令を使用する可能性の高いアプリケーションは複雑であるか、マルチスレッドされたアプリケーションであるか、または高度に最適化された固有に作成したロックおよび同期化手法を使用しているデバイス・ドライバです。

B.1 必要なコード・チェック

OpenVMS では、21264 プロセッサで実行されるコードにこのようなシーケンスがないかどうか確認してください。プロセス間ロック、マルチスレッド、プロセッサ間通信を行うコードでは、特に注意する必要があります。

Alpha 実行可能プログラムを分析して、規則に準拠していないコード・シーケンスがあるかどうか調べるために、SRM_CHECK ツールが開発されました。このツールは、エラーが発生する可能性のあるシーケンスを検出し、エラーを報告し、問題のあるシーケンスのマシン・コードを表示します。

B.2 コード分析ツール (SRM_CHECK) の使用

SRM_CHECK ツールは、OpenVMS Alpha Version 7.3-2 Operating System の次の場所にあります。

```
SYS$SYSTEM:SRM_CHECK.EXE
```

SRM_CHECK ツールを実行するには、フォーリン・コマンドとして定義 (または DCL\$PATH 機能を使用) し、チェックするイメージの名前を指定して起動します。問題が検出されると、マシン・コードが表示され、イメージ情報の一部が印刷されます。次の例では、このツールを使用して myimage.exe というイメージを分析する方法を示しています。

```
$ define DCL$PATH []  
$ srm_check myimage.exe
```

このツールでは、ワイルドカード検索がサポートされます。ワイルドカード検索を開始するには、次のコマンド行を使用します。

```
$ srm_check [*...]* -log
```

チェックされたイメージのリストを作成するには、-log 修飾子を指定します。-output 修飾子を使用すれば、出力をデータ・ファイルに書き込むことができます。たとえば、次のコマンドは出力を CHECK.DAT という名前のファイルに書き込みます。

```
$ srm_check 'file' -output check.dat
```

このイメージの MAP ファイルを調べれば、ツールからの出力を使用して、このシーケンスを生成したモジュールを検索することができます。表示されるアドレスは、MAP ファイルから見つけることができるアドレスに直接対応しています。

次の例に、SYSTEM_SYNCHRONIZATION.EXE というイメージに対して、分析ツールを使用した結果できる出力を示します。

```
** Potential Alpha Architecture Violation(s) found in file...  
** Found an unexpected ldq at 00003618  
0000360C AD970130 ldq_l R12, 0x130(R23)  
00003610 4596000A and R12, R22, R10  
00003614 F5400006 bne R10, 00003630  
00003618 A54B0000 ldq R10, (R11)  
Image Name: SYSTEM_SYNCHRONIZATION  
Image Ident: X-3  
Link Time: 5-NOV-1998 22:55:58.10  
Build Ident: X6P7-SSB-0000  
Header Size: 584  
Image Section: 0, vbn: 3, va: 0x0, flags: RESIDENT EXE (0x880)
```

system_synchronization.exe の MAP ファイルには、次の情報が格納されます。

```
EXEC$NONPAGED_CODE      00000000 0000B317 0000B318 (    45848.) 2 ** 5
SMPROUT                  00000000 000047BB 000047BC (    18364.) 2 ** 5
SMPINITIAL               000047C0 000061E7 00001A28 (     6696.) 2 ** 5
```

アドレス 360C は SMPROUT モジュールにあり、0 ~ 47BB のアドレスが格納されています。モジュールから出力されたマシン・コードを確認することで、コードの位置を調べ、リスト行番号を使用して、対応するソース・コードを識別することができます。SMPROUT のベースが 0 以外の場合は、アドレス(この場合は 360C) からベースを減算して、リスト・ファイル内での相対アドレスを求める必要があります。

このツールは、可能性のある違反を出力の中で報告します。SRM_CHECK は通常、セクションの属性によってイメージ内のコード・セクションを識別することができますが、OpenVMS イメージの場合は、同じ属性を持つデータ・セクションが格納されることがあります。この結果、SRM_CHECK はデータをコードであるかのようにスキャンし、データ・ブロックを規則に準拠していないコード・シーケンスであると解釈することがあります。このような状況はあまり発生することがなく、MAP とリスト・ファイルを調べることで検出できます。

B.3 規則に準拠していないコードの特徴

SRM_CHECK ツールによって検出される規則に準拠しないコードは、次の 4 つに分類できます。これらの大部分は、新しいコンパイラで再コンパイルすることで修正できます。ソース・コードを変更しなければならないことがありますが、そのような場合はまれです。コンパイラのバージョンの詳細については、第 B.5 節を参照してください。

- OpenVMS コンパイラの一部のバージョンでは、「ループ・ローテーション」と呼ばれる最適化中に規則に準拠しないコード・シーケンスが発生します。この問題が発生するのは、ASM 関数を使用して C/C++ ソースに埋め込まれているアセンブリ言語コード内で LD_x_L/ST_x_C 命令を使用する C または C++ プログラムの場合か、MACRO-32 または MACRO-64 で作成されたアセンブリ言語の場合だけです。LD_x_L 命令と ST_x_C 命令の間に分岐が発生していることもありました。

この問題は再コンパイルすることで対処できます。

- 非常に古い BLISS, MACRO-32, DEC Pascal, または DEC COBOL コンパイラでコンパイルされた一部のコードには、規則に準拠しないシーケンスが含まれていることがあります。これらのコンパイラの初期のバージョンには、コード・スケジューリングのバグがあり、load_locked の後にロードが誤ってスケジューリングされていました。

この問題は再コンパイルすることで対処できます。

- ごくまれに、空きレジスタの数が少なすぎる場合には、MACRO-32 コンパイラは BBSSI 命令または BBCCI 命令に対して、規則に準拠しないコード・シーケンスを生成することがあります。

この問題は再コンパイルすることで対処できます。

- MACRO-64 または MACRO-32 が誤ってコーディングされ、アセンブリ言語が ASM 関数を使用して C または C++ ソースに組み込まれ、誤ってコーディングされたために、エラーが発生することがあります。

この問題が発生する場合は、ソース・コードを変更しなければなりません。新しい MACRO-32 コンパイラは、規則に準拠しないコードに対して、コンパイル時にフラグを付けます。

SRM_CHECK ツールがイメージから違反を検出した場合は、適切なコンパイラを使用してイメージを再コンパイルしなければなりません (第 B.5 節を参照)。再コンパイルした後、イメージを再び分析する必要があります。再コンパイルの後にも違反が発生する場合は、ソース・コードを調べ、コード・スケジューリング違反が発生する原因を追求しなければなりません。その後、必要に応じてソース・コードを変更します。

B.4 コーディングの必要条件

『Alpha Architecture Reference Manual』では、プロセッサ間のデータの不可分な更新を実行する方法を説明しています。特に『Third Edition』には、この問題に関するさらに詳しい情報が含まれています。また、インターロックされたメモリ・シーケンスの規則について詳しく説明されています。

次の 2 つの必要条件が満たされない場合は、規則に準拠しないコードが生成されます。

- インターロックされたシーケンスで、LDx_L (load locked) 命令と STx_C (store conditional) 命令の間にメモリ操作 (load または store) を指定できない。
- LDx_L 命令と STx_C 命令の間で分岐を実行できない。このような場合は、分岐を実行せずに、LDx_L から STx_C に「フォール・スルー」しなければならない。

ターゲットが LDx_L とそれに対応する STx_C の間にある分岐を実行すると、規則に準拠しないシーケンスが作成される。たとえば、次の例で "label" への分岐を実行すると、分岐命令自体がシーケンスの内部にあるか外部にあるかにかかわらず、規則に準拠しないコードが作成される。

```
LDx_L Rx, n(Ry)
...
label: ...
STx_C Rx, n(Ry)
```


したがって、SRM_CHECK ツールは次の条件を検索します。

- LDx_L と STx_C の間のメモリ操作 (LDx/STx)
- LDx_L と STx_C の間に宛先がある分岐
- 先行する LDx_L 命令のない STx_C 命令

これは通常、LDx_L から STx_C へ逆方向分岐が実行されることを示します。ただし、デバイス・メールボックス書き込みを実行するハードウェア・デバイス・ドライバは例外です。これらのドライバは、STx_C を使用してメールボックスに書き込みを実行します。この状況は初期の Alpha システムでのみ検出され、PCI ベースのシステムでは検出されません。

- LDx_L と STxC の間にある余分な命令

AARM では、LDx_L と STx_C の間の命令の数を 40 未満にすることを推奨しています。理論的には、40 より多くの命令があると、シーケンスが完了しないようにするためにハードウェア割り込みが発生します。しかし、実際にはこの状況が発生したという報告はありません。

次の例に、SRM_CHECK でフラグが付けられたコードを示します。

```
** Found an unexpected ldq at 0008291C
00082914 AC300000 ldq_l R1, (R16)
00082918 2284FFEC lda R20, 0xFFEC(R4)
0008291C A6A20038 ldq R21, 0x38(R2)
```

この例では、LDQ 命令が LDQ_L の後、STQ_C の前に検出されています。LDQ は、再コンパイルまたはソース・コードの変更によって、このシーケンスの外部に移動しなければなりません (第 B.3 節を参照してください)。

```
** Backward branch from 000405B0 to a STx_C sequence at 0004059C
00040598 C3E00003 br R31, 000405A8
0004059C 47F20400 bis R31, R18, R0
000405A0 B8100000 stl_c R0, (R16)
000405A4 F4000003 bne R0, 000405B4
000405A8 A8300000 ldl_l R1, (R16)
000405AC 40310DA0 cmple R1, R17, R0
000405B0 F41FFFFA bne R0, 0004059C
```

この例では、LDL_L と STQ_C の間から分岐が検出されています。この場合、LDx_L と STx_C の間に、アーキテクチャで要求されている「フォール・スルー」パスがありません。

注意

LDx_L から STx_C へのこの逆向きの分岐は、「ループ・ローテーション」最適化によって発生する、規則に準拠しないコードの特徴です。

次の MACRO-32 ソース・コードは「フォール・スルー」パスがあるものの、ロック・シーケンス内に可能性のある分岐とメモリ参照があるために、規則に準拠しないコードを示しています。

```
getlck: evax_ldq1  r0, lockdata(r8) ; Get the lock data
        movl      index, r2        ; and the current index.
        tstl      r0                ; If the lock is zero,
        beql      is_clear         ; skip ahead to store.
        movl      r3, r2           ; Else, set special index.
is_clear:
        incl      r0                ; Increment lock count
        evax_stqc r0, lockdata(r8) ; and store it.
        tstl      r0                ; Did store succeed?
        beql      getlck           ; Retry if not.
```

このコードを修正するには、INDEX の値を読み込むためのメモリ・アクセスを最初に LDQ_L/STQ_C シーケンスの外側に移動しなければなりません。次に、ラベル IS_CLEAR への、LDQ_L と STQ_C の間の分岐を取り除かなければなりません。この場合、CMOVEQ 命令を使用して分岐を取り除くことができます。CMOVxx 命令はしばしば、単純な値の移動の周囲にある分岐を取り除くために役立っています。次の例に、修正されたコードを示します。

```
        movl      index, r2        ; Get the current index
getlck: evax_ldq1  r0, lockdata(r8) ; and then the lock data.
        evax_cmoveq r0, r3, r2     ; If zero, use special index.
        incl      r0                ; Increment lock count
        evax_stqc r0, lockdata(r8) ; and store it.
        tstl      r0                ; Did write succeed?
        beql      getlck           ; Retry if not.
```

B.5 コンパイラのバージョン

表 B-1 は、規則に準拠しないコード・シーケンスを生成する可能性のあるコンパイラのバージョンと、再コンパイルするときに使用する推奨最小バージョンについて説明します。

表 B-1 OpenVMS コンパイラのバージョン

古いバージョン	推奨最小バージョン
BLISS V1.1	BLISS V1.3
DEC Ada V3.5	HP Ada V3.5A
DEC C V5.x	DEC C V6.0
DEC C++ V5.x	DEC C++ V6.0
DEC COBOL V2.4 , V2.5 , V2.6	DEC COBOL V2.8

(次ページに続く)

表 B-1 (続き) OpenVMS コンパイラのバージョン

古いバージョン	推奨最小バージョン
DEC Pascal V5.0-2	DEC Pascal V5.1-11
MACRO-32 V3.0	OpenVMS Version 7.1-2 については V3.1 OpenVMS Version 7.2 については V4.1
MACRO-64 V1.2	下記参照

MACRO-64 アセンブラの現在のバージョンでも、ループ回転の問題が発生することがあります。しかし、MACRO-64 ではデフォルトでコードの最適化が実行されないため、この問題が発生するのは、最適化が有効に設定されている場合だけです。SRM_CHECK が MACRO-64 コードから規則に準拠しないシーケンスを検出した場合は、最初に最適化せずに再コンパイルする必要があります。その後、再びテストしてもシーケンスにフラグが付けられる場合は、ソース・コード自体に修正の必要な非準拠シーケンスが含まれています。

21264 プロセッサのある Alpha コンピュータでは、『Alpha Architecture Reference Manual, Third Edition』に記載されている LDx_L 命令および STx_C 命令のインターロックされたメモリ・シーケンスの制限事項を厳密に守る必要があります。インターロックされたメモリ命令の使用がアーキテクチャのガイドラインに準拠するように、MACRO-32 for OpenVMS Alpha Version 3.1 コンパイラに、追加のチェックが盛り込まれました。

『Alpha Architecture Reference Manual, Third Edition』の 4.2.4 項に、インターロックされたメモリ・シーケンス内での命令の使用規則が説明されています。MACRO-32 for OpenVMS Alpha Version 3.1 コンパイラは、MACRO-32 ソース・コードから生成するコード内でこれらの規則に従います。ただし、このコンパイラは EVAX_LQxL および EVAX_STxC の組み込みを用意しているため、これらの命令を直接ソース・コードに記述することができます。

MACRO-32 for OpenVMS Alpha Version 4.1 コンパイラは、準拠していないコード・シーケンスを検出するために追加のコード・チェックを行い、警告メッセージを表示するようになりました。

B.6 ALONONPAGED_INLINE または LAL_REMOVE_FIRST によるコードの再コンパイル

OpenVMS Alpha の MACRO-32 コードのうち、SYS\$LIBRARY:LIB.MLB マクロ・ライブラリから ALONONPAGED_INLINE マクロまたは LAL_REMOVE_FIRST マクロを起動するコードは、OpenVMS Version 7.2 以降で再コンパイルして、これらのマクロの正しいバージョンが取得されるようにしなければなりません。これらのマクロを変更すると、新しい Alpha 21264 (EV6) 以上のプロセッサで検出される可能性のある、同期化に関する問題を修正できます。

注意

EXE\$ALONONPAGED ルーチン (またはその変形) を呼び出すソース・モジュールは、再コンパイルする必要がありません。これらのモジュールは、ユーザに意識させることなく、このリリースに含まれているルーチンの正しいバージョンを使用します。

A

Ada コンパイラ	
I64 ではまだ利用できない	5-8
AlphaServer 2100	
SCSI コントローラの制限事項	6-4
コンソール表示	6-3
AlphaServer 4100	
FRU テーブルの制限事項	6-4
AlphaServer 8200 システム	
FRU テーブルの制限事項	6-4
AlphaServer 8400 システム	
FRU テーブルの制限事項	6-4
AlphaServer ES47/ES80/GS1280 システム	
MBM での時刻の設定	6-6
RAD のサポート	6-5
STOP/CPU およびシャットダウン動作	6-5
システム・パラメータ PHYSICAL_MEMORY の設定	6-5
ソフト・パーティションでの INIT コンソール・コマンドの使用	6-4
ライセンス要件	6-5
AlphaServer GS シリーズ・システム	
デバイスの制限事項	6-6
複数 I/O ポートの制限事項	6-9
AlphaStation 200/400	
ISA_CONFIG.DAT の変更が必要	6-9
AlphaStation 255	
PCI 構成の制限事項	6-10
ANALYZE/DISK/SHADOW コマンド	4-33
異種デバイス・シャドウイング	4-34
ANALYZE/ERROR_LOG コマンド	
I64 に移植されない	3-3
ATI RADEON 7000 グラフィック	6-10
ハードウェア・アクセラレーション 3D グラフィックはサポートされていない	6-11
ATI RADEON 7500 グラフィック	6-11 ~ 6-19
DECW\$OPENGLSHR_RADEON.EXE のリネーム	6-14
DECwindows サーバがハンガアップ	6-15
OpenGL は IEEE 演算のみをサポート	6-15
高リフレッシュ・レートでの画像への影響	6-15
性能改善	
DMA 操作	6-13
OpenGL テクスチャ・コピー	6-13
テキストその他のビットマップの描画	6-12

ATI RADEON 7500 グラフィック (続き)	
ハードウェア・アクセラレーションによるインダイレクト 3D レンダリング	6-13
問題の修正	6-13
Authorize コーティリティ	
DEFAULT と SYSTEM アカウントの新規クォータ	4-1
AUTOGEN	
NEWPARAMS.DAT ファイル	4-2

B

Backup API	
ジャーナリング・イベント	5-8
BASIC	
STARLET ライブラリの作成に V1.5A が必要	2-2
BLISS	
HP BLISS コンパイラを参照	
BMC コンソールの制約事項 (I64 のみ)	6-1
BUGCHECKFATAL システム・パラメータ	5-17

C

C++ コンパイラ	
HP C++ コンパイラを参照	
CANCEL SELECTIVE 関数, LTDRIVER での使用の改善	5-37
CDSA	5-15
Secure Delivery ADK	5-15
CLUE コマンド	
I64 に移植されていない	5-41
CLUSTER_CONFIG.COM	
ルート・ディレクトリ名の制限	4-14
CMAP ファイル	
新規	2-2
COB\$SWITCHES	5-21
COBOL RTL	
HP COBOL RTL を参照	
COM for OpenVMS	
アプリケーションの重負荷によるエラー	2-3
サポート	2-3
Common Data Security Architecture	
CDSA を参照	
Compaq Open3D レイヤード・プロダクト	
V8.2 ではサポートされない	A-1

CONVERT-I-SEQ エラー	
CONVERT/NOSORT	5-17
CPUSPINWAIT バグ・チェック	5-17
CRCTX ルーチンの機能の強化	6-31
CREATE/MAILBOX コマンド	
一時的な制約	3-3
C RTL	
lseekの問題	5-12
C RTL	5-9~5-14
DECC\$SHR_EV56 のリンクの問題を修正	
正	5-14
<errno.h>内の新しい EABANDONED コード	5-12
getcとgetcharの接頭辞の問題	5-11
getcの問題	5-11
getpwnam_r, getpwuid_rのポインタの問題	5-10
globの問題の修正	5-14
iconvプロトタイプの変更	5-10
IPV6 構造体がパック形式に	5-13
mktimeの問題の修正	5-12
mmap, mprotectの変更	5-9
_PAL_BUGCHK の問題の修正	5-13
<poll.h>の問題	5-13
snprintfによるバッファの上書き	5-9
statvfsの問題の修正	5-13
<stddef.h>の修正	5-11
std名前空間の問題	5-11
_strtok_r32と_strtok_r64がスコープ内になった	5-10
v*scanfと*snprintfの名前空間の問題	5-11
vsnprintfによるバッファの上書き	5-9
メモリ・リーク	5-9
Ctrl/H キー・シーケンス	
DEL への再マッピング (I64 のみ)	6-2
C コンパイラ	
HP C コンパイラを参照	
C プログラム	
コンパイルと、大文字と小文字の区別	5-8
C ランタイム・ライブラリ	
C RTL を参照	
D	
data-reduced ELF オブジェクト・ライブラリ	
リンク対象	5-34
DCE RPC	
HP DCE RPC for OpenVMS を参照	
DCL コマンド	3-3
DCL コマンド・プロシージャ	
SET SHADOW と SHOW SHADOW の使用	4-30
DDB 構造体	
アップデート	5-4
Debugger	
OpenVMS Debugger を参照	
DECamds	
V8.2 ではサポートされない	A-2
DECC\$SHR_EV56 のリンクの問題を修正	5-14
DECdfs for OpenVMS	
Version 2.4 が必要	2-3
DECdtm	
Oracle 8i, 9i	4-3
DECevent	A-3
サポートされない	A-2
DECforms Web Connector	
OpenVMS Version 7.3-1 以降で実行	2-3
Decimal Support RTL	
HP Decimal Support RTL を参照	
DECmigrate	
V8.2 Open Source Tools CD-ROM に無い	3-4
DECnet for OpenVMS	1-3
DECnet/OSI	
DECnet-Plus for OpenVMS を参照	
DECnet-Plus for OpenVMS	1-3
新しいバージョンが必要	1-16
DEC PL/I	2-4
DECram	
HP DECram を参照	
DECGRAM	
DECRYPT コマンドとの競合	2-9
DECwindows Motif	
HP DECwindows Motif を参照	
DECwindows X11 ディスプレイ・サーバ	
グラフィック・ボードのサポート	6-19~6-20
周辺デバイスの接続要件	1-11
DECwrite	
サポートの終了	A-3
Delta/XDelta デバッグ	5-18
Delta は I64 では利用できない	5-18
I64 システムでの XDelta の制限事項	5-18
XDelta のプラットフォームによる相違点	5-19
レジスタ表示に関する考慮	5-19
DEVICE_NAMING システム・パラメータ	4-3
DEVOFFLINE エラー	
ANALYZE/DISK/SHADOW コマンド	4-33
DIAGNOSE コマンド	
サポートされない	3-3, A-2
DIGITAL Modular Computing Components (DMCC)	
KZPDA コントローラと PBXGA グラフィック・カード	6-20
SRM コンソールの更新	6-20
Digital Personal Workstation	6-21
DIRECTORY コマンド	
出力の変更	4-20
DSSI ディスク・デバイス	
マイクロコード・リビジョン・レベル	6-24
DUMP_DEV エラー (I64 のみ)	4-5

E

ECP (Enterprise Capacity and Performance)	4-4
EDIT/FDL	
推奨パッケージ・サイズの変更	4-4
EFI\$CP ユーティリティ	
使用は推奨できない	4-4
EFI ツール	
VMS_SHOW DUMP_DEV エラー (I64 のみ)	4-5
ELV	
Error Log Viewer (ELV) ユーティリティを参照	
Encryption for OpenVMS Alpha	1-14
Enterprise Capacity and Performance (ECP)	
ECP を参照	
<errno.h>内の EABANDONED コード	5-12
Error Log Report Formatter (ERF)	
サポート終了	A-3
Error Log Viewer (ELV) ユーティリティ	4-5
EV6 Alpha プロセッサ	B-1

F

Fast Path	
無効化, ES40 上の Galaxy	4-19
Fibre Channel	
互換キット	4-12
テープ・デバイスのマルチパス・フェールオーバーの制限事項	4-17
FMS キット	2-4
Fortran	
HP Fortran を参照	
Freeware	3-1
OpenVMS I64 でメニューが利用できない	3-1

G

Galaxy	
定義	4-18
getcとgetcharの接頭辞の問題	5-11
getcの問題	5-11
getpwnam_r, getpwuid_rのポインタの問題	5-10
Gigabit Ethernet スイッチの制限事項	4-16
globの問題の修正	5-14
Graphical Configuration Manager (GCM)	
Galaxy でのサポート	4-18
Graphical Configuration Utility (GCU)	4-18

H

HP BLISS コンパイラ	
規則に準拠していないコードを実行した結果	B-1
警告 (I64 のみ)	5-20
HP C++ コンパイラ	
規則に準拠していないコードを実行した結果	B-1
HP COBOL RTL	5-21
SET および COB\$SWITCHES	5-21
HP C コンパイラ	
規則に準拠していないコードを実行した結果	B-1
HP DCE RPC for OpenVMS	2-5 ~ 2-8
HP Decimal Support RTL	5-21
HP DECram	2-8
DECram と DECRYPT のコマンド競合	2-9
V8.2 では SIP として出荷	2-8
Version 2.5 (VAX のみ)	2-8
アップグレード前の削除 (Alpha のみ)	1-14
最大ディスク・サイズ	2-9
HP DECwindows Motif	
LAT トランスポート・インタフェースのサポート	2-11
一時停止画面でロック解除できないことがある (Alpha のみ)	2-10
キーボードのサポートの制限 (I64 のみ)	1-10
サポートされているバージョン	1-5
スタートアップ・メッセージ	1-11
ユーザが作成したトランスポートのサポート	2-12
利用可能な言語バリエーション	2-11
HP Fortran	
I64 用	5-22
HP Secure Web Browser	
必要メモリ量の増加	3-4
HP Secure Web Browser	
ODS-2 でインストール・エラーになる (I64 のみ)	3-5
HP Secure Web Server	
サポート	2-12
HP SSL のインストール	1-5
HSGnn	
障害	6-21
Hypersort ユーティリティ	5-26 ~ 5-28

I

I64 システムでのリポート	1-10
iconv	
修正	3-4
iconvプロトタイプの変更	5-10
IDE CD-ROM	6-21
INITIALIZE コマンド	
/CLUSTER_SIZE が指定されないときの新しい動作	4-7

INITIALIZE コマンド (続き)	
新しいV/GPT 修飾子	4-9
新規の/ERASE キーワード	4-7
INIT コンソール・コマンド	
ES47/ES80/GS1280 ソフト・パーティションでの 使用	6-4
INSTALL コーティリティ	4-7
Integrity サーバ	
構成	1-6
ファームウェア	1-7
Intel@アセンブラ (I64 のみ)	5-28
Invocation context block	5-50
IPL 要件	
VMS フォーク・スレッドの作成	5-6
IPV6 構造体がパック形式に	5-13
ISA_CONFIG.DAT	
将来のリリースでのサポートの中止	A-4

K

Kerberos	1-15
アップグレード前の V1.0 の削除	1-15
KPB 拡張	5-3

L

LANCP	
アップグレード後にデバイス・データベースを変 換する (Alpha のみ)	1-16
LDAP API の問題点	5-30
LIB\$I64_GET_FR	5-50
LIB\$I64_GET_GR	5-50
LIB\$I64_PUT_INVO_REGISTERS	5-50
LIB\$I64_SET_FR	5-50
LIB\$I64_SET_GR	5-50
LIB\$LOCK_IMAGE	
ヘルプからの漏れ	5-50
LIBOTS2	
HP Decimal Support RTL を参照	
Librarian コーティリティ	
オブジェクト・モジュール名の長さの問題 (I64 の み)	5-30
Librarian コーティリティ	5-28
.STB ファイルの制限事項 (I64 のみ)	5-29
エラー報告に関する問題	5-29
LIBRTL	
呼び出し標準ルーチン (I64 のみ)	5-50
LINK_ORDER ELF セクション・ヘッダ・フラ グ	5-34
Linker マップ	
Image Synopsis セクションの誤り	5-35
LINK/NATIVE_ONLY のヘルプ・テキストの明確 化	5-31
Logical Disk (LD) コーティリティ	
問題の修正	4-9
lseekの問題	5-12
LTDRIVER の制限事項	5-37

M

MACRO-32 コンパイラ	
規則に準拠していないコードを実行した結 果	B-1
コードの再コンパイル	B-7
MACRO-64 アセンブラ	
規則に準拠していないコードを実行した結 果	B-1
Mail コーティリティ (MAIL)	
呼び出し可能メールがカーネル・スレッドで使用 された場合の問題	5-37
MEDOFLL エラー	
ANALYZE/DISK/SHADOW コマンド	4-33
mktimeの問題の修正	5-12
mmap, mprotectの変更	5-9
MMG_CTLFLAGS システム・パラメータ	4-26
MP コンソールの制約事項 (I64 のみ)	6-1
MULTIPROCESSING システム・パラメー タ	5-17

N

NEWPARAMS.DAT ファイル	
AUTOGEN	4-2

O

On-Disk Structure	
レイアウトの変更	4-9
Open3D グラフィック	
コントローラ・ボードのサポート	6-28
ライセンス方式の変更	6-22
OpenVMS Alpha 用 Linker	5-30 ~ 5-33
LINK/NATIVE_ONLY のヘルプ・テキストの明確 化	5-31
RMS_RELATED_CONTEXT オプショ ン	5-31
スタックのエレメント数は最大 25 に制 限	5-33
多数のファイルの処理時にハングアップす る	5-31
ライブラリ・チェックの動作の変更	5-33
OpenVMS Cluster システム	4-10 ~ 4-17
CI と LAN との間の切り替えによる性能の低 下	4-15
Gigabit Ethernet スイッチの制限事項	4-16
SCSI マルチパス・フェイルオーバ	4-13
互換キット	4-12
複合バージョン用の互換キット	4-12
ルート・ディレクトリ名の制限	4-14
ローリング・アップグレード	1-13
OpenVMS Debugger	5-38
Alpha	
以前のバージョンはサポートされな い	5-40
Alpha および I64	

OpenVMS Debugger	
Alpha および I64 (続き)	
SET SCOPE コマンドの変更	5-40
SHOW IMAGE コマンドの変更	5-40
I64	
Basic 言語での問題	5-39
C++ 言語での問題	5-39
COBOL 言語での問題	5-39
Fortran 言語での問題	5-39
Pascal 言語での問題	5-40
全般的な状況と回避方法	5-38
OpenVMS Galaxy	4-18 ~ 4-19
および ES40	
Fast Path の無効化	4-19
非圧縮ダンプの制限事項	4-19
ライセンスの実行	6-6
OpenVMS I64	
DVD からのブート	1-10
OpenVMS I64 用 Linker	5-33 ~ 5-37
data-reduced ELF オブジェクト・ライブラリ	5-34
DIFTYPE および RELODIFTYPE メッセージ	5-35
LINK_ORDER セクション・ヘッダ・フラグ	5-34
LINK/NATIVE_ONLY のヘルプ・テキストの明確化	5-31
OpenVMS Alpha 用 Linker との違い	5-34
マップに誤り	5-35
OpenVMS Management Station	4-19
OpenVMS Registry	
Version 2 フォーマットのデータベースの破壊	4-19
OpenVMS システム・ダンプ・アナライザ	
CLUE コマンドは I64 に移植されていない	5-41
READ コマンドのデフォルトの変更	5-41
SHOW CALL_FRAME 機能の変更 (I64 のみ)	5-41
OpenVMS 用 HP MACRO	5-23 ~ 5-26
CODGENWARN メッセージ (Alpha のみ)	5-24
I64 システム上	5-23
INSV 命令でメモリを余計に上書きする (I64 のみ)	5-25
/OPTIMIZE=VAXREGS 修飾子は I64 ではサポートされない	5-23
/TIE 修飾子の Alpha と I64 でのデフォルト値	5-23
整数の除算で“V”コンディション・コードが正しくセットされない (I64 のみ)	5-25
整数を負の最大値で除算するとコンパイラがクラッシュする (I64 のみ)	5-24
操作単位のサポート (I64 のみ)	5-24
デフォルトでは整数のゼロ除算エラーは検出されない (I64 のみ)	5-24
浮動小数点数のゼロ除算エラーが検出されない (I64 のみ)	5-25

OpenVMS 用 MACRO	
OpenVMS 用 HP MACRO を参照	
OpenVMS 呼び出し標準規則	5-15

P

_PAL_BUGCHK	
問題の修正	5-13
Pascal	
STARLET ライブラリの作成には V5.8A が必要 (Alpha のみ)	2-13
アップグレード後の再インストール (Alpha)	2-13
PATHWORKS for OpenVMS (Advanced Server)	
ACE の表示	4-20
PCB\$T_TERMINAL	
サイズの拡張	5-5
PCI 構成の制限事項	6-10
PE1 システム・パラメータ	4-8
PEdriver	
LAN 輻輳への対処	4-15
PGFLQUOTA の問題	5-29
PL/I	
I64 に含まれないライブラリ	5-42
RTL サポート	2-4
<poll.h>の問題	5-13
POOLCHECK システム・パラメータ	5-17
POSIX スレッド・ライブラリ	5-43 ~ 5-47
I64 での THREADCP コマンドの動作	5-44
POSIX 1003.4a Draft 4 インタフェースのサポート中止	A-4
デバッグ計測機能は動作しない	5-47
動的 CPU 構成	5-46
浮動小数点例外 (I64 のみ)	5-44
例外処理中のスタック・オーバフロー (I64 のみ)	5-43
PowerStorm 300/350 PCI グラフィック・サポート	6-23
Open3D のライセンスはチェックされない	6-23

R

RF73 および RFnn ディスク, コントローラ・メモリ・エラー	6-24
RMS Journaling	5-47
ジャーナル・ファイル作成の変更	5-48
順方向ジャーナリング	5-49
リカバリ・ユニット・ジャーナル・ファイルのリモート・アクセス	5-49
RU ジャーナリング	
リカバリ・ユニット・ジャーナリングを参照	
RZnn ディスク・ドライブ	6-26 ~ 6-28

S

SCD	
System Code Debugger を参照	
SCSI コントローラ	
AlphaServer 2100 システムでの制限事項	
項	6-4
SCSI デバイス・ドライバ	6-29
SCSI マルチパスの非互換性	4-13
SDA	
OpenVMS のシステム・ダンプ・アナライザを参照	
SET DEVICE/SWITCH コマンド	4-17
SET PASSWORD コマンド	4-6
SET SHADOW	
DCL コマンド・プロシージャでの使用	4-30
SHAD_MAX_UNIT	
メモリ消費量	1-17
SHADOW_MAX_UNIT	
デフォルト設定	1-17
SHOW LICENSE	
/HIERARCHY には SYSLOCK 特権が必要 (I64 のみ)	3-3
/OE には SYSLOCK 特権が必要 (I64 のみ)	3-3
SHOW SHADOW	
DCL コマンド・プロシージャでの使用	4-30
Smart Array 5300	
Volume Shadowing の制限事項	4-32
SMG\$	
ドキュメントの修正	5-51
SMHANDLER	
サーバ管理プロセスを参照	
snprintfによるバッファの上書き	5-9
Software Public Rollout Reports	2-1
SORT32 ユーティリティ	5-28, 5-52 ~ 5-53
SPLINVIPL バグ・チェック	5-7
SRM_CHECK ツール	B-2
SSL のインストール	1-5
statvfsの問題の修正	5-13
<stddef.h>の修正	5-11
std名前空間の問題	5-11
_strtok_r32と_strtok_r64がスコープ内になった	5-10
SYS\$ACM	
I64 での使用	5-55
SYS\$STARLET_C.TLB	
アップグレード・エラー	1-13
SYSGEN	
セキュリティ監査の修正	4-21
SYSMAN	
DUMP_PRIORITY	4-21
SYSTEM_CHECK システム・パラメータ	5-17
System Code Debugger	
I64 で使用できない	5-53
System Event Analyzer (SEA) ユーティリティ	
I64 でのサポート	2-14

System Event Log (SEL)	
Integrity サーバ上でのクリア	1-7

T

TCP/IP Services for OpenVMS	1-3
TECO エディタ	
I64 システムでは利用できない	3-5
Terminal Fallback Facility (TFF)	4-26
制限事項	4-27
TFF	
Terminal Fallback Facility を参照	
THREADCP コマンド	
I64 での動作	5-44
TQE	
タイマ・キュー・エントリを参照	
Traceback 機能	
問題の修正	5-57

U

UCB 構造体	
アップデート	5-4

V

v*scanfと*snprintfの名前空間の問題	5-11
VAX Cluster キャッシュ	
Virtual I/O キャッシュを参照	
VCC	
Virtual I/O キャッシュを参照	
VFC 形式の順編成ファイル	5-49
VIOC	
Virtual I/O キャッシュを参照	
Virtual I/O キャッシュ (VIOC)	
I64 では利用不可	4-29
XFC に置き換え	4-29
VMS_SHOW	
DUMP_DEV エラー (I64 のみ)	4-5
Volume Shadowing for OpenVMS	4-29 ~ 4-35
ANALYZE/DISK/SHADOW コマンド	4-33
DDS	4-34
/MINICOPY を使用したディスマウントの問題	4-35
Smart Array 5300 (KZPDC) の制限事項	4-32
異種デバイス・シャドウイング (DDS)	
KZPDC の制限事項	4-32
注意事項	4-30
互換キット	4-12
デバイス名の必要条件	4-29
マージ操作の性能	4-33
vsprintfによるバッファの上書き	5-9

W

Watchpoint コーティリティ	5-57
WEBES	
I64 でのサポート	2-14

X

X.25 データ・リンクがサポートされていない (I64 のみ)	2-8
XA	4-3
XFC	
拡張ファイル・キャッシュを参照	

Z

zic のアップデート	5-14
ZLX グラフィック・ボードのサポート	6-28

ア

アーカイブされたマニュアル	A-4
アップグレード	
V7.3-1 からのアップグレードでのエラー (Alpha のみ)	1-13
パス	1-12

イ

異種デバイス・シャドウイング (DDS)	
ANALYZE/DISK/SHADOW コマンドの動作	4-34
KZPDC の制限事項	4-32
書き込みビットマップとの相互作用	4-30
異種デバイス・シャドウイング (DDS)	
Smart Array 5300 の制限事項	4-32
インストール・エラー	
HP Secure Web Browser	3-5
インストールとアップグレードの情報	
ネットワーク・オプション	1-3
インターロックされたメモリ命令の使用	B-1

オ

オンライン・ヘルプ	
ヘルプを参照	

カ

回線切り替え	
性能の低下	4-15
外部認証	4-5
I64 サポート	4-5
SET PASSWORD コマンド	4-6
パスワードの有効期限切れの通知	4-6
書き込みビットマップ	
異種デバイス・シャドウイング (DDS)	4-30

拡張 DDT ビット	
修正された問題点	5-37
拡張ファイル・キャッシュ (XFC)	4-29
カーネル・スレッド	
リカバリ・ユニット・ジャーナリングと互換性がない	5-47
関連製品	
Software Public Rollout Reports	2-1
現在のリリースでサポートされるバージョン	2-1

キ

規則に準拠していないコード	B-1, B-3
---------------	----------

ク

クラスタ	
OpenVMS Cluster システムを参照	
クラスタ互換キット	4-12
グラフィック	
I64 システムでのサポート	6-11
グラフィック・ボードのサポート	6-28

ケ

現在のリリースでのアプリケーションのサポート	2-1
------------------------	-----

コ

高速ロック再マスタリング	4-8
コンパイラ	
規則に準拠していないコード	B-1, B-6

サ

削除キー	
再マッピングが必要 (I64 のみ)	6-2
サーバ管理プロセス (SMHANDLER)	4-20

シ

システム・サービス	
\$GETJPI の項目コード SCHED_CLASS_NAME の記述誤り	5-53
I64 システムでの SYS\$ACM の使用	5-55
PFN マップ・セクション	
新しい項目コードが必要 (I64 のみ)	5-54
キャッシュされないメモリ (I64 のみ)	5-54
SYS\$GOTO_UNWIND	5-55
SYS\$GOTO_UNWIND_64	5-55
システム・ディスク	
古いシステムと互換性がない	1-4
システムのクラッシュ	
回復 (I64 のみ)	4-1

システムのハングアップ	
回復 (I64 のみ)	4-1
システム・パラメータ	4-22 ~ 4-26
BUGCHECKFATAL	5-17
DEVICE_NAMING	
デバイス・ユニット最大数の増加に使	
用	4-3
MMG_CTLFLAGS ドキュメントの誤り ..	4-26
MULTIPROCESSING	5-17
PE1	4-8
PHYSICAL_MEMORY	6-5
POOLCHECK	5-17
SYSTEM_CHECK	5-17
新しいパラメータ	4-22
廃止されたパラメータ	4-22
廃止されたものの表示	4-23
変更	4-23
修正キット	
入手方法	1-3
複合バージョンの OpenVMS Cluster システムで	
必要	4-12
順方向ジャーナリング	5-49

ス

スレッド単位のセキュリティ	
デバイス・ドライバへの影響	5-5
特権付きコードへの影響	5-5

セ

セキュリティ監査の修正	4-21
-------------------	------

ソ

ソフトウェアのサポート方針	1-1
---------------------	-----

タ

タイマ・キュー・エントリ (TQE)	5-56
タイム・ゾーンの変更	4-28

テ

デバイス・ドライバ	
IPL の設定	6-30
MON バージョンの処理	6-30
SCSI	6-29
再コンパイルと再リンク	6-29 ~ 6-30
スレッド単位のセキュリティの影響	6-30
テープ・ロボット	
自動マルチパス・フェールオーバ	4-17
デュアル・コントローラ HSGnn	
障害	6-21

ト

統合されたグラフィック・ボード		6-20
動的 CPU 構成		
POSIX スレッド・ライブラリ	5-46	
ドキュメントの変更と修正		
\$GETJPI ヘルプの誤り	5-53	
LIB\$ヘルプの漏れ	5-50	
『OpenVMS RTL Screen Management (SMG\$)		
Manual』	5-51	
『OpenVMS System Services Reference		
Manual』	5-53	
アーカイブされたマニュアル	A-4	
ドキュメントの変更と訂正		
LINK/NATIVE_ONLY のヘルプ・テキストの明確		
化	5-31	
『OpenVMS Performance Management』		
.....	4-26	
『OpenVMS システム管理ユーティリティ・リフ		
アレンス・マニュアル』	4-26	
オンライン・ヘルプ	4-26	
『OpenVMS Cluster システム』	4-14	
特権データ構造体		
CPU の名前空間	5-3	
KPB 拡張	5-3	
PCB\$T_TERMINAL のサイズの拡張	5-5	
UCB と DDB のアップデート	5-4	
アップデート	5-5	
スレッド単位のセキュリティの影響	5-5	
動的スピンロックのフォーク	5-3	
64 ビットの論理ブロック番号	5-3	
変更	5-2	
トレースバック機能		
API エラー (I64 のみ)	5-56	

ネ

ネットワーク・オプション	1-3
--------------------	-----

ハ

バックアップ・ユーティリティ		
動作変更	4-3	
パッチ・キット		
複合バージョンの OpenVMS Cluster システムで		
必要	4-12	
パーティション		
ソフト	4-18	
ハード	4-18	
ハード・パーティション	4-18	

ヒ

- ビジュアル・スレッド
V8.2 ではサポートされない A-4

フ

- ファームウェア
Alpha サーバ用 1-11
Integrity サーバ用 1-7
- 浮動小数点型データ
アプリケーションへの考慮 5-7
- フリーウェア A-1
- プログラム全体の浮動小数点モード (I64 のみ) 5-57
- プログラムの再コンパイル
Alpha の場合 5-1
I64 の場合 5-1

ヘ

- ヘルプ
変更 3-2

ホ

- ポート・ドライバ\$QIO
制限事項 5-37

マ

- マイクロコード・リビジョン・レベル
DSSI ディスク・デバイス 6-24
更新するためのコマンド 6-25
- マニュアルの変更と訂正
『Guide to OpenVMS File Applications』
. 5-19
- マルチパス・フェールオーバ
Fibre Channel テープ・デバイスの制限事
項 4-17

- テープ・ロボット 4-17

メ

- メモリ・リーク 5-9

ヨ

- 呼び出し標準規則
OpenVMS 呼び出し標準規則を参照

ラ

- ライセンスについて 6-6 ~ 6-9
- ライブラリアン・ユーティリティ
data-reduced ELF オブジェクト・ライブラリと
のリンク (I64 での制限事項) 5-28

リ

- リカバリ・ユニット・ジャーナリング
カーネル・スレッド 5-47
ジャーナリングされたファイルへのリモート・ア
クセス 5-48
制限事項 5-49
ファイル作成の変更 5-48
- リタイア製品情報 A-1 ~ A-5

ロ

- ロック・マネージャ
高速ロック再マスタリング 4-8
ロック値ブロックの拡張 4-8
- ロケール
新規 2-11
- ローテートするレジスタ 5-50
- および ICB 5-15
- およびメカニズム・ベクタ 5-15

HP OpenVMS V8.2 リリース・ノート【翻訳版】

2005年4月 発行

日本ヒューレット・パカード株式会社

〒140-8641 東京都品川区東品川 2-2-24 天王洲セントラルタワー

電話 (03)5463-6600 (大代表)

BA322-90010

