

Tru64 UNIX

Tru64 UNIX 概要

Part Number: AA-RK3KE-TE

2002 年 11 月

ソフトウェア・バージョン: Tru64 UNIX Version 5.1B 以上

本書は、HP Tru64 UNIX オペレーティング・システムで提供する機能について概要を説明しています。

日本ヒューレット・パッカード株式会社

© 2002 日本ヒューレット・パッカード株式会社

本書の著作権は日本ヒューレット・パッカード株式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・パッカードの文書による許可なしに、その全体または一部を、いかなる場合にも再版あるいは複製することを禁じます。

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

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

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

COMPAQ, Compaq ロゴ, Digital ロゴは U.S. Patent and Trademark Office に登録されています。Alpha, AlphaServer, NonStop, TruCluster, および Tru64 は米国 Compaq Computer Corporation の商標です。

Microsoft, Windows および Windows NT は米国 Microsoft 社の登録商標です。Intel は米国 Intel 社の登録商標です。Motif, OSF/1, UNIX, The Open Group および X/Open は、The Open Group の米国ならびに他の国における商標です。

このドキュメントに記載されているその他の会社名および製品名は、各社の商標または登録商標です。

本書に記載されているその他の製品名は、各社の商標または登録商標です。

原典: Technical Overview (AA-RH8XE-TE)
© 2002 Hewlett-Packard Company

目次

まえがき

1 Tru64 UNIX の概要

1.1	概要	1-1
1.1.1	Tru64 UNIX の沿革	1-2
1.1.2	標準	1-3
1.2	Tru64 UNIX の機能と拡張機能	1-4
1.2.1	スケーラビリティ	1-5
1.2.2	性能	1-6
1.2.3	可用性と保守性	1-7
1.2.4	インストール方法の種類	1-8
1.2.5	システムおよびネットワーク管理機能	1-9
1.2.6	ARMTech リソース管理ソフトウェア	1-14
1.2.7	UNIX と Microsoft Windows の相互運用性	1-14
1.2.8	Advanced Printing Software	1-14
1.2.9	System V のサポート	1-15
1.2.10	Secure Web Server	1-15
1.2.11	Tru64 UNIX のフリーウェア	1-16
1.2.12	ドキュメント	1-16
1.3	オプション・コンポーネント	1-18
1.3.1	TruCluster Server	1-18
1.3.2	Logical Storage Manager	1-19
1.3.3	Advanced File System Utilities	1-19
1.3.4	Developer's Toolkit	1-20
1.3.5	Advanced Server for UNIX	1-20
1.3.6	Multimedia Services	1-20

1.3.7	その他のソフトウェア	1-21
1.3.7.1	Internet Express for Tru64 UNIX	1-21
1.3.7.2	Enterprise Toolkit for Visual Studio	1-22
1.3.7.3	ソース・マテリアル・オプション	1-22
1.4	パッケージの内容	1-22
1.4.1	マニュアル	1-23
1.4.2	ライセンス	1-23
2	システム管理	
2.1	インストール	2-1
2.1.1	フル・インストール	2-2
2.1.2	アップデート・インストール	2-3
2.1.3	インストールのクローニング	2-6
2.1.4	構成のクローニング	2-7
2.1.5	クラスタのローリング・アップグレード	2-8
2.1.6	setld ユーティリティ	2-8
2.1.7	インストール関連のドキュメント	2-9
2.2	システム・パッチの適用	2-9
2.3	システムの設定	2-11
2.4	Logical Storage Manager	2-16
2.5	システム管理ユーティリティ	2-18
2.5.1	SysMan Menu	2-19
2.5.2	SysMan Station	2-22
2.5.3	CDE アプリケーション・マネージャ	2-23
2.6	性能とイベントの管理	2-26
2.6.1	HP Insight Manager	2-26
2.6.2	Compaq Analyze	2-27
2.6.3	Monitoring Performance History ユーティリティ	2-27
2.6.4	sys_check ユーティリティ	2-27
2.6.5	X ベースのユーティリティ	2-28

2.6.6	環境モニタリング	2-29
2.6.7	イベント・マネージャ	2-29
2.6.8	DECevent Translation and Reporting ユーティリティ	2-30
2.7	クラッシュ・ダンプの管理	2-31
2.8	ハードウェア管理	2-32
2.8.1	hwmgr ユーティリティ	2-32
2.8.2	Dynamic Device Recognition	2-33
2.9	動的にロード可能なサブシステム	2-33
2.10	動的システム構成	2-34
2.11	データレス管理サービス	2-34
3	ネットワーク機能	
3.1	インターネット・プロトコル	3-1
3.1.1	アプリケーション・レベルのプロトコル	3-3
3.1.1.1	DNS プロトコル	3-4
3.1.1.2	ルーティング・プロトコル	3-4
3.1.1.3	ファイル転送プロトコル (FTP)	3-9
3.1.1.4	UDP トランスポート上のネットワーク・ファイル・システム・プロトコル	3-9
3.1.1.5	TCP トランスポート上のネットワーク・ファイル・システム・プロトコル	3-9
3.1.1.6	TELNET プロトコル	3-10
3.1.1.7	簡易ファイル転送プロトコル (TFTP)	3-10
3.1.1.8	FINGER プロトコル	3-10
3.1.1.9	シンプル・メール転送プロトコル (SMTP)	3-11
3.1.1.10	シンプル・ネットワーク管理プロトコル (SNMP)	3-11
3.1.1.11	POP3	3-11
3.1.1.12	IMAP4	3-12
3.1.1.13	RSVP	3-12
3.1.2	トランスポート・レベルのプロトコル	3-13

3.1.2.1	ユーザ・データグラム・プロトコル (UDP)	3-13
3.1.2.2	伝送制御プロトコル (TCP)	3-14
3.1.3	インターネット・ネットワーク・レベルのプロトコル	3-15
3.1.3.1	IPv4	3-15
3.1.3.2	IPv6	3-17
3.1.3.3	IPsec (Internet Protocol Security)	3-19
3.1.3.4	アドレス解決プロトコル (ARP)	3-21
3.1.3.5	インターネット制御メッセージ・プロトコル (ICMP) .	3-22
3.2	ネットワーク・インタフェース層	3-23
3.2.1	ATM	3-23
3.2.2	Ethernet ネットワーク	3-24
3.2.3	Fast Ethernet ネットワーク	3-24
3.2.4	Gigabit Ethernet ネットワーク	3-24
3.2.5	FDDI ネットワーク	3-24
3.2.6	トークン・リング・ネットワーク	3-25
3.2.7	シリアル・ライン IP (SLIP) および 圧縮シリアル・ライン IP (CSLIP)	3-25
3.2.8	PPP (Point-to-Point Protocol)	3-25
3.2.9	多重アダプタのサポート	3-26
3.2.10	リンク・アグリゲーション	3-26
3.2.11	NetRAIN	3-27
3.2.12	仮想ローカル・エリア・ネットワーク	3-28
3.3	アプリケーション・プログラミング・インタフェース	3-28
3.3.1	X/Open トランスポート・インタフェース (XTI)	3-29
3.3.2	ソケット	3-31
3.3.3	STREAMS	3-31
3.3.4	ソケットと STREAMS の相互作用	3-32
3.3.5	データ・リンク・プロバイダ・インタフェース (DLPI)	3-32
3.3.6	拡張可能 SNMP インタフェース (eSNMP)	3-32
3.4	ネットワーク管理ソフトウェア	3-33

3.4.1	ネットワーク・コマンドおよびユーティリティ	3-33
3.4.2	Ethernet パケット・フィルタおよびパケット・フィルタ・アプリケーション	3-34
3.4.3	DHCP	3-35
3.4.4	インターネット・ブート・プロトコル・デーモン (bootpd)	3-36
3.4.5	SNMP エージェント	3-37
3.4.6	gated デーモン	3-37
3.4.7	screend デーモン	3-39
3.4.8	UUCP	3-39
3.4.9	ローカル・エリア・トランスポート (LAT)	3-40
3.4.10	ネットワーク・インタフェースの監視	3-40
3.5	ネーミング・サービス	3-41
3.5.1	ドメイン・ネーム・サービス	3-41
3.5.2	ネットワーク情報サービス (NIS)	3-42
3.6	タイム・サービス	3-42
3.6.1	ネットワーク・タイム・プロトコル (NTP)	3-43
3.6.2	時刻同期化プロトコル (TSP)	3-44

4 ファイル・システム

4.1	概要	4-1
4.2	仮想ファイル・システム	4-2
4.3	Advanced File System	4-3
4.4	UNIX ファイル・システム	4-4
4.5	クラスタ・ファイル・システム	4-4
4.6	NFS	4-5
4.6.1	NFS Version 3 の機能	4-6
4.6.2	NFS に対する拡張	4-7
4.7	CD-ROM ファイル・システム (CDFS)	4-10
4.8	DVD ファイル・システム	4-10
4.9	メモリ・ファイル・システム(MFS)	4-11

4.10	/proc ファイル・システム	4-11
4.11	File-on-File マウント・ファイル・システム	4-12
4.12	ファイル記述子ファイル・システム(FDFS)	4-13
 5 カーネル, シンメトリック・マルチプロセッシング, NUMA, 仮想メモリおよびデバイスのサポート		
5.1	Tru64 UNIX カーネル	5-1
5.1.1	カーネルのチューニング	5-1
5.1.2	拡張されたカーネル・デバッグ機能	5-2
5.2	対称型マルチプロセッシング	5-3
5.3	不均等メモリ・アクセス (NUMA)	5-5
5.4	仮想メモリ	5-6
5.4.1	ページの管理と追跡	5-7
5.4.2	変更されたページのプリライト	5-8
5.4.3	属性を使用したページングとスワッピングの制御	5-8
5.4.4	ページング操作	5-9
5.4.5	スワッピング操作	5-10
5.4.6	スワップ領域割り当てモード	5-11
5.4.7	スワップ・バッファの使用	5-12
5.4.8	ユニファイド・バッファ・キャッシュ	5-12
5.5	デバイスのサポート	5-12
 6 開発環境		
6.1	Compaq C コンパイラ	6-2
6.2	デバugg	6-3
6.2.1	dbx デバugg	6-3
6.2.2	ladebug デバugg	6-3
6.3	プロファイリング・ツール	6-4
6.4	シェアード・ライブラリ	6-6

6.4.1	クイックスタート	6-6
6.4.2	ダイナミック・ローダ	6-6
6.4.3	バージョニング	6-7
6.5	実行時ライブラリ	6-7
6.6	Java 開発キット	6-8
6.7	開発コマンド	6-8
6.8	スレッドのサポート	6-9
6.8.1	POSIX スレッド・ライブラリ	6-9
6.8.2	Visual Threads	6-9
6.8.3	TIS (Thread Independent Services)	6-10
6.9	メモリ・マップ・ファイル・サポート(mmap)	6-10
6.10	リアルタイム・ユーザとプログラミング環境	6-10
6.11	ネットワーク・プログラミング・インタフェース	6-11
7	セキュリティ	
7.1	概要	7-1
7.2	識別, 認証, 承認	7-2
7.3	任意アクセス制御	7-3
7.4	監査サブシステム	7-3
7.5	オブジェクト再使用	7-4
7.6	トラステッド・コンポーネントのセキュリティ	7-4
7.7	完全性	7-5
8	国際化機能	
8.1	国際化機能の概要	8-1
8.2	サポート言語	8-3
8.3	ロケールの作成	8-7
8.4	コードセット変換	8-7
8.5	Unicode および dense コード・ロケール	8-8

8.6	Unicode サポート	8-9
8.7	国際化ソフトウェア構成ユーティリティ	8-9
8.8	ユーロ文字のサポート	8-10
8.9	dxim 入力サーバ	8-11
8.10	国際化 Curses ライブラリ	8-12
8.11	その他の国際化機能	8-12
9	ウィンドウ環境	
9.1	Common Desktop Environment	9-1
9.2	X ウィンドウ・システム	9-3
9.2.1	X クライアント・ライブラリ	9-3
9.2.2	X サーバ	9-4
9.2.3	マルチヘッド・グラフィック・サポート	9-4
9.2.4	X サーバ拡張機能	9-4
9.2.5	ディスプレイ・マネージャ	9-8
9.2.6	フォント・サーバ	9-9
9.2.7	X クライアント	9-9
9.3	Motif コンポーネント	9-10
9.3.1	拡張ウィジェット・セット	9-10
9.3.2	X クライアント	9-11
10	UNIX と Windows 間の相互運用性	
10.1	Advanced Server for UNIX	10-1
10.2	データ・アクセス (ODBC および JDBC)	10-2
A	インターネット標準	
A.1	RFC 標準	A-1
A.2	RFC 以外の標準	A-11

UNIX および一般コンピュータの用語集

索引

図

2-1	システム・セットアップ	2-13
2-2	クイック・セットアップ	2-14
2-3	カスタム・セットアップ	2-15
2-4	SysMan Menu	2-20
2-5	SysMan Station	2-23
3-1	TCP/IP プロトコル	3-3
3-2	XTI , STREAMS , およびソケット間の相互関係	3-30
3-3	DHCP 構成	3-36
4-1	ファイル・システム	4-2

表

2-1	ユーザが制御するアップデート・プロセスの機能	2-5
2-2	アップデート・プロセスの内蔵機能	2-6
8-1	言語とロケール	8-3
A-1	RFC 標準	A-1



まえがき

本書では、HP Tru64 UNIX オペレーティング・システムおよびそのコンポーネントで提供する機能について概要を説明します。

本書の対象読者

本書は、Tru64 UNIX の機能について関心のあるすべてのユーザを対象としています。

本書の構成

本書の構成は以下のとおりです。

- | | |
|--------|---|
| 第 1 章 | Tru64 UNIX オペレーティング・システムの概要について説明します。 |
| 第 2 章 | システム管理のさまざまな機能を説明します。 |
| 第 3 章 | 利用できるネットワーク・プロトコルおよびアプリケーションについて説明します。 |
| 第 4 章 | Tru64 UNIX がサポートするいくつかのファイル・システムについて説明します。 |
| 第 5 章 | シンメトリック・マルチプロセッシング、仮想メモリ、NUMA、およびデバイスのサポートなど、カーネルに関連する問題について説明します。 |
| 第 6 章 | Tru64 UNIX 開発環境によって提供される主な機能について説明します。 |
| 第 7 章 | Tru64 UNIX が満たす C2 評価クラスの必要条件やその他のセキュリティ機能について説明します。 |
| 第 8 章 | 国際化機能について説明します。 |
| 第 9 章 | Tru64 UNIX のウィンドウ環境である Common Desktop Environment (CDE) および X Window System Motif の構成要素について説明します。 |
| 第 10 章 | UNIX システムと Windows NT とのシームレスな対話を可能にする Tru64 UNIX アプリケーションについて説明します。 |

付録 A	Tru64 UNIX が準拠している RFC および RFC 以外の標準の一覧です。
用語一覧	Tru64 UNIX 環境で使用される UNIX 用語と一般的なコンピュータ用語について説明します。

関連資料

- QuickSpecs には、弊社の製品についての技術情報が記載されています。 *Tru64 UNIX Operating System QuickSpecs* と、個別にライセンスされるソフトウェア・コンポーネントの QuickSpecs は、次の URL で参照できます。

<http://tru64unix.compaq.co.jp/document/quickspecs.html>

- Tru64 UNIX のドキュメント・セットは、Tru64 UNIX Documentation CD-ROM および Web で入手できます。ドキュメント・セットについての情報は、『ドキュメント概要』を参照してください。Web 上のドキュメントは、次の URL で参照できます。

<http://tru64unix.compaq.co.jp/document/index.html>

- 本書では、インターネット・プロトコルおよびポリシを定義しているさまざまな RFC (Request for Comments) を参照しています。RFC は、たとえば以下に示すサイトをはじめ、いくつかの Web サイトで公開されています。

<http://www.ietf.cnri.reston.va.us/home.html>

<http://www.cis.ohio-state.edu/hypertext/information/rfc.html>

インターネットに接続されているシステム上で、本書をオンラインで参照している場合は、RFC をクリックして、Web ブラウザ内にその RFC のテキストを表示することができます。

- ActiveAnswers によって、Web ベースの包括的なソリューション・プログラムが提供されます。このプログラムには、主要なアプリケーションの計画、配備、操作、保守のためのユーティリティが含まれています。これらのユーティリティは、マネジメント、データベースおよびビジネス・アプリケーション、高度な可用性、およびインターネット・ソリューションをはじめ、さまざまな領域で、アプリケーションを補助します。

ActiveAnswers は、ソリューションに要する時間を短縮し、同時に HP をエンタープライズ・ソリューション・プロバイダの中心的存在にすることを主な目的としています。また、Microsoft、SAP、Novell、

Oracle , Genesys など , 主要な ISV とのパートナーシップを結んでいます。顧客とシステム・インテグレータのための , Web ベースのソリューションも開発されています。次の Web サイトにある ActiveAnswers の Web ページを参照してください。

<http://www.compaq.com/activeanswers/>

- コンパックのホームページは , 次のサイトにあります。

<http://www.compaq.co.jp/>

- Tru64 UNIX のホームページは , 次のサイトにあります。

<http://www.compaq.co.jp/tru64unix/>

本書の表記法

本書では , 以下の表記法を使用しています。

file イタリック体 (斜体) は , 変数値 , プレースホルダ , および関数の引数名を示します。

cat(1) リファレンス・ページの参照には , 該当するセクション番号をカッコ内に示します。たとえば , cat(1) は , cat コマンドについての情報が , リファレンス・ページのセクション 1 に記載されていることを示します。

MB/s この記号は M バイト/秒を意味します。

Mb/s この記号は M ビット/秒を意味します。

用語一覧で使われている用語 本書のオンライン・バージョンでは , さまざまな用語が『UNIX および一般コンピュータの用語集』にリンクされています。用語をクリックすると , その定義が表示されます。ブラウザの Back ボタンをクリックすれば , 元の画面に簡単に戻ることができます。



Tru64 UNIX の概要

この章では、Tru64 UNIX オペレーティング・システムを紹介し、その機能の概要を説明します。この章に記載されている構成要素の多くは、後の章でさらに詳しく説明しています。この章では、次のトピックについて説明します。

- オペレーティング・システムの主要な機能、OSF/1 誕生以来の歴史、このシステムが準拠している規格の要約 (1.1 節)
- エンタープライズ・アプリケーション、ハイパフォーマンス・テクニカル・コンピューティング (HPTC)、およびインターネット通信を支援する Tru64 UNIX の機能 (1.2 節)
- コンピューティング環境の要求を満たす Tru64 UNIX のオプション・コンポーネント (1.3 節)
- オペレーティング・システムの構成要素とメディア・キットの内容 (1.4 節)

1.1 概要

HP Tru64 UNIX は、現在購入できる製品の中で、最も長く使用されているフル 64 ビット・オペレーティング・システムです。以下のような機能で、最高レベルの性能、スケーラビリティ、および可用性を提供するとともに、システムの管理や操作は簡単に行うことができます。

- 優れた管理インタフェースを提供することにより、システム管理作業が簡素化されています。
- システムのインストレーション、セットアップ、および管理の複雑さが大幅に緩和されています。
- 使い慣れたインタフェース、一貫したルック・アンド・フィールのユーティリティ群、および処理の自動化により、ユーザのトレーニングの必要性が最小限に抑えられ、さらにはトータルな所有コストの削減につながります。

- 数テラバイト規模のシステム構成に対応しており，ビジネス・ニーズの増大に柔軟に対応できます。
- ファイル・システム，ストレージ管理，およびシステム・ネットワークに関する性能が向上しています。
- UNIX と Windows 2000 の高度な統合を実現しています。
- TruCluster Server のサポートにより，可用性に対するユーザの厳しいニーズに対応しています。

Tru64 UNIX は，明快で簡潔なシステム管理環境で次のような機能を実現しています。

- システム管理作業の簡素化
- システム・ファイルを上書きしないアップデート・インストレーション，およびシステムを迅速に起動し実行することができるフル・インストレーションを GUI で実行することができます。
- 他社製のディスクおよびグラフィック・カードのローダブル・ブート・パス・サポートを含む，ローダブル・ドライバやその他のカーネル・サブシステムのサポート
- ディスク，テープ，および CD-ROM のような SCSI ストレージ・デバイスを実質的に自動管理できます。

Tru64 UNIX は，リアルタイム・サポート，SMP (Symmetrical Multiprocessing)，データレス・サーバ/クライアントをサポートしています。また，シェアード・ライブラリ，スレッド，メモリ・マップ・ファイルなどを使用するアプリケーションの開発を支援するための，非常に多くの機能を提供しています。Tru64 UNIX は，統一 UNIX 仕様，X/Open UNIX ブランド，POSIX 1003.1b (リアルタイム)，および POSIX 1003.1c (DECthreads と呼ばれる POSIX スレッド) に完全に準拠しています。

共通デスクトップ環境 (CDE) は，Tru64 UNIX の標準のユーザ・インタフェースです。

1.1.1 Tru64 UNIX の沿革

Tru64 UNIX は，OSF (Open Software Foundation) V1.0 および V1.2 の機能と Motif V1.2.5 のグラフィカル・ユーザ・インタフェース，およびプログラミング環境を実装した HP のオペレーティング・システムです。また

Tru64 UNIX は、X11R6.5 (X Window System, Version 11, Release 6.5) の機能を完全にサポートしています。

Tru64 UNIX オペレーティング・システムは、カーネギー・メロン大学の Mach V2.5 カーネル・デザインと BSD (Berkeley Software Distribution) V4.3 および V4.4、UNIX System Laboratories の System V Release 4.0、その他のソフトウェアのソース・コード、パブリック・ドメイン、および Hewlett-Packard Company の拡張機能などのコンポーネントをベースに設計されているマルチユーザ/マルチタスクの 64 ビット先進カーネル・アーキテクチャです。

Tru64 UNIX の以前のバージョンは、DIGITAL UNIX と呼ばれていました。初期出荷時は DEC OSF/1 という名前でした。

1.1.2 標準

Tru64 UNIX は UNIX 98 製品標準に準拠しています。UNIX 98 製品標準は UNIX 95 製品標準から大幅に拡張されています。

UNIX 98 では、次の機能が拡張されています。

- スレッド・インタフェース
- マルチバイト・サポート拡張 (MSE)
- ラージ・ファイルのサポート
- ダイナミック・リンク
- ハードウェア・データ長への依存性および制限を取り除くための変更

UNIX 98 は、次の製品標準からなっています。

- 国際化システム・コールおよびライブラリ拡張 V2
- コマンドおよびユーティリティ V3
- C 言語
- ソケット V2
- トランスポート・サービス (XTI) V2
- 国際化端末インタフェース (XCurses)

Tru64 UNIX の UNIX 98 Conformance Statement Questionnaire (適合性に関する質問) は、次の URL の The Open Group の Web サイトにあります。

<http://www.opengroup.org/csq/>

Tru64 UNIX では、ソフトウェア管理機能や、リアルタイム・サポート用の API 群などの、UNIX 98 製品標準のオプションの拡張はサポートしていません。

このバージョンの Tru64 UNIX は、UNIX 98 ワークステーション製品標準にも準拠しています。この製品標準は UNIX 98 製品標準とほぼ同じですが、共通デスクトップ環境製品標準に準拠しなければならないという追加要件があります。

CDE 標準は、X ウィンドウ・システムをサポートしているシステム用の共通のグラフィカル・ユーザ・インタフェース環境である、X/Open 共通デスクトップ環境を定義しています。この標準は、準拠システムがサポートしなければならない、標準アプリケーション・プログラミング・インタフェース、コマンド行動作、データ交換フォーマット、およびプロトコルだけではなく、機能やサポートするインフラストラクチャの標準セットを定義しています。この標準は、グラフィカル・ユーザ・インタフェース環境が通常備えている機能 (ウィンドウ、ウィンドウ管理、セッション管理、ファイル管理、電子メール、テキスト編集、カレンダーおよびスケジュール管理、電卓、アプリケーション・ビルダと統合サービス、プリント・ジョブ・サービス、およびヘルプ・サービスなど) の標準形式を定義しています。

このオペレーティング・システムは、Berkeley 4.3 および System V プログラミング・インタフェースと互換性があります。また、System V Interface Definition (SVID3 Base and Kernel Extensions) でコンパイルすることにより、System V アプリケーションをサポートすることもできます。

Tru64 UNIX がサポートしている標準についての詳細は、付録 A および『*Tru64 UNIX Operating System QuickSpecs*』を参照してください。

1.2 Tru64 UNIX の機能と拡張機能

Tru64 UNIX によって、エンタープライズ・アプリケーション、ハイパフォーマンス・テクニカル・コンピューティング (HPTC)、ビジネス・インテリジェンス、およびインターネット通信をサポートする多数の重要な機能と拡張機能が提供されます。これらの機能の大部分は基本オペレーティング・システムに統合されていますが、一部の機能は「Tru64 UNIX Associated Products」CD-ROM に含まれているオプション・コンポーネントおよび別

ライセンスの製品で提供されています。Tru64 UNIX のメディア・キットの詳細な説明は、1.4 節を参照してください。

以降の項では、オペレーティング・システムの主要な機能について説明します。

1.2.1 スケーラビリティ

Tru64 UNIX のスケーラビリティ機能としてクロック周波数の異なる CPU の混在および NUMA アーキテクチャがサポートされています。

- クロック周波数の異なる CPU の混在のサポート

クロック周波数が異なる CPU の混在のサポートにより、現在装備されているプロセッサの速度をあまり気にせずに、AlphaServer GS160、および GS320 システムのプロセッサを追加購入できます。AlphaServer GS シリーズのコンピュータは、CPU の数を 1 ~ 32 の範囲で簡単に変更でき、最大 256GB のメモリが搭載可能で、異なるオペレーティング・システムの複数のインスタンスを実行することができます。

- NUMA (不均等メモリ・アクセス)

AlphaServer GS80、GS160、GS320、ES80、および GS1280 プラットフォームでは、不均等メモリ・アクセス (NUMA) アーキテクチャを採用しており、多数の CPU を搭載したマルチプロセッサ環境を可能にしています。オペレーティング・システム・ソフトウェアも NUMA に対応しており、従来のアプリケーションをソース・コードの変更なしで、最適な性能で動作させることができます。

また、NUMA プラットフォーム上でより直接的な CPU 管理やメモリ・アクセスを必要とする大規模で複雑なアプリケーションでは、NUMA API を使用できます。

NUMA についての詳細は、5.3 節を参照してください。

- Capacity on Demand

Capacity on Demand プログラムを使用すると、将来の拡張に備えて、前もって CPU を追加したシステムを注文しておき、処理能力の拡張が必要になった時点でその分の代金を支払って使用することができます。拡張用 CPU は最初から装着されているため、システムをリブートすることなしに拡張用 CPU を有効にできます。

Capacity on Demand についての詳細情報および Capacity on Demand キットは次の Web サイトからダウンロードできます。

http://www.compaq.com/hps/promotions/gs_capacity.html

1.2.2 性能

Tru64 UNIX では、性能に関する機能としてビッグ・ページ、パラレル・バス・スキャンニング、VLM (Very Large Memory) をサポートします。

- ビッグ・ページのサポート

Tru64 UNIX V5.1B では、プロセスごとのビッグ・ページのサポートを提供します。これにより、変換バッファ・ミスによる性能上の不利益を最小限に抑え、アプリケーションの性能を改善することができます。最大 4MB (512 - 8KB ページ) のページ・サイズがサポートされます。

この機能により、アプリケーションの変更や再ビルドを行わないで、より大きなページ・サイズを利用することができます。ビッグ・ページ・メモリ割り当てには次のような特性があります。

- プロセスのアドレス・スペースにおける仮想ページを、システムの物理メモリがサポートする最も適切なマルチ・ページ密度にマップすることができます。
- メモリ割り当て要求に対してページ・サイズの拡張を行うのが適切かどうかは、メモリ・タイプごとに設定されたしきい値によって判断されます。

省略時の設定では、可変メモリ・ページ・サイズの使用は無効になっています。この機能を有効にする方法については、`sys_attrs_vm(5)` を参照してください。

- パラレル・スキャンニング

パラレル・スキャンニングは、デバイスの認識に必要な時間を短縮することによりブート時間を短縮する Tru64 UNIX Version 5.1B の新機能です。SCSI および Fibre Channel バスのパラレル・スキャンニングを有効にすると、システムはすべてのバスの走査を (1 つずつ順番にではなく) 同時に行います。この機能により、デバイスを検出する時間を短縮できます。多数のストレージ・デバイスを持つシステムでは、この機能によるブート時間の短縮は非常に大きなものになります。

この機能は省略時の設定では無効になっています。この機能を使用するための設定については『ハードウェア管理ガイド』を参照してください。

- VLM (Very Large Memory) と大規模ストレージ機能

Tru64 UNIX は、最高 28 GB までのメモリをサポートするため、VLM 構成においてメモリをより効率的に利用できます。4 TB 以上のファイル・システムおよびストレージ・システムがサポートされます。

- 動的チューニング

ほとんどのチューニング作業は、システムをリブートすることなくシステムの稼働中に行うことができます。

1.2.3 可用性と保守性

Tru64 UNIX は、ハードウェア・コンポーネントの追加、取り外しなどを支援する保守機能を提供します。

- オンラインでの CPU の追加/交換機能 (OLAR: Online Addition and Removal)

OLAR 技術により、AlphaServer GS160 および GS320 システムの CPU をシステムを停止させることなく交換したりアップグレードすることができます。「CPU ホット・スワップ」とも呼ばれるこの新しい機能を使用すると、システムを動作させたままで、CPU を高速なものに交換したり、故障または誤動作している CPU を交換することができます。

Compaq Analyze ユーティリティには、交換が望ましい CPU を検出し、オペレーティング・システムに通知する機能が追加されました。交換すべき CPU のオフライン化を自動的に要求するようにオペレーティング・システムを構成することもできます。この機能により、障害が発生した CPU が原因でシステム全体が停止する前に、問題を含む CPU を無効にできるため、システムの連続稼働時間を長く保つことができます。この OLAR 通知機能を使用するためには、WEBES (Web-Based Enterprises Services) キットから Compaq Analyze をインストールする必要があります。

OLAR 機能は、システムの状態を集中管理するためのシステム管理アプリケーション、SysMan ユーティリティ (2.5 節を参照) に統合されています。

OLAR については、『*Managing Online Addition and Removal*』を参照してください。

- デバイスのサポート
 - オペレーティング・システムがサポートしている SCSI-3 標準により、SCSI バスごとに最高 256 のターゲット・アドレスおよび LUN アドレスが使用でき、デュアル・パス (デバイスへの代替パス) がサポートされています。
 - デバイス・ネーミング・モデルに柔軟性があり、SCSI バスごとに 256 を超えるデバイス名を使用できます。このモデルは、SCSI-3 および Fibre Channel をサポートしています。
 - Tru64 UNIX は、Fibre Channel スイッチ接続、および適応負荷分散機能 (adaptive load balancing) を備えたマルチ・コンカレント・パス (64) をサポートしています。また、SCSI-2 と SCSI-3 の両方のコマンドをサポートしています。

1.2.4 インストレーション方法の種類

Tru64 UNIX は、いくつかの方法でインストールできます。さまざまなユーティリティを使用し、そのプロセスをできるだけ単純にすることができます。Tru64 UNIX で用意されているインストール方法を以下に説明します。各インストレーション方法については、2.1 節でさらに詳しく説明しています。

- フル・インストレーション

オペレーティング・システムを新規にインストールします。フル・インストレーションでは、新しいファイル・システムとスワップ領域が作成され、それらがインストールされるディスク・パーティション上の既存のシステムおよびユーザ作成ファイルは上書きされます。このインストールを実行した後、通常運用のためのシステム設定が必要です。2.1.1 項を参照してください。

- アップデート・インストレーション

Version 5.1 または Version 5.1A がインストールされているシステムを Version 5.1B にアップデートします。このタイプのインストールでは、システム上で実行されたディスク・パーティション、ファイル・システム、ファイルのカスタマイズ、ネットワークと印刷の環境、ユーザ・アカウント、ユーザ作成ファイル、およびその他のシステムの設定が維持されます。2.1.2 項を参照してください。

- クローン・インストレーション

すでにインストールされたモデル・システムから同一または類似のハードウェアで構成された 1 つ以上のシステムへ、フル・インストールを複製できます。インストールのクローニングは、同一タイプの多数のシステムにインストールを行なう場合、理想的な方法です。クローニングによって、同一のシステム・インストールを複製することができ、システムごとにフル・インストールを実行せずに済みます。 2.1.3 項を参照してください。

- 構成のクローニング

フル・インストールの際、すでに構成されているシステムから、ネットワーク環境および印刷環境を複製できます。 2.1.4 項を参照してください。

- ローリング・アップグレード

TruCluster Server ソフトウェアは、クラスタ環境においてローリング・アップグレードをサポートしています。これにより、クラスタ内の各メンバ・システムのソフトウェアのアップグレードをクラスタが機能している状態で行うことができます。サービスにアクセスしているクライアントは、ローリング・アップグレードが進行中であることは意識せずに引き続きサービスを受けることができます。 2.1.5 項 および 2.2 節 を参照してください。

1.2.5 システムおよびネットワーク管理機能

Tru64 UNIX は、システムの設定、構成、およびチューニングを容易にし、日常の保守および管理タスクを単純化するアプリケーションや機能を数多く提供しています。

以下に、システムおよびネットワーク管理機能について概要を説明します。第 2 章には、さらに詳しい説明があります。

- SysMan Menu によるタスクの管理

さまざまなシステム管理タスクを編成するためのフレームワークを提供する SysMan Menu により、種々の管理アプリケーションをこのユーティリティから起動することができます。

SysMan Menu は、CDE、HTML、ASCII テキスト環境のいずれかで実行できます。すなわち、SysMan Menu で提供するすべての作業は、X11 をサポートするディスプレイ、Microsoft Windows、Linux、または

Mac OS が実行されているパソコンや文字セル端末から実行できます。
SysMan Menu についての詳細は、2.5.1 項を参照してください。

- SysMan Station によるタスクの管理

SysMan Station は、Tru64 UNIX システムをグラフィカルに表示し、任意のコンピュータからシステムを遠隔管理するための機能です。この Java ユーティリティは、TruCluster Server ソフトウェアに完全に統合されています。SysMan Station についての詳細は、2.5.2 項を参照してください。

- Quick Setup によるシステムの構成

重要かつ一般的に実行されるシステム設定の手順を提供するための機能です。この機能により、システムを基本的なシステム構成でセットアップするための、迅速かつユーザ・フレンドリな方法が提供されます。さらに細かな設定を行いたい場合は、カスタム・セットアップ・アプリケーションを使用します。キャラクタ・セル・インタフェースでアクセスできるクイック・セットアップ・アプリケーションは、システム・セットアップ・アプリケーション (/usr/sbin/setup コマンド) から利用できます。詳細については 2.3 節を参照してください。

- メモリ巡視機能

メモリ巡視機能 (memory trolling) とは、メモリ・エラーを事前に発見して対処するために、システム・メモリを読み込むプロセスです。

この機能は、ユーザが有効/無効を切り替えることができ、
vm_troll_percent 属性により巡視速度を調整することができます。この巡視速度はいつでも変更できます。メモリ巡視機能についての詳細は、『*Managing Online Addition and Removal*』および memory_trolling(5) リファレンス・ページを参照してください。

- NetWork Setup ウィザードによるネットワークへの接続

Network Setup ウィザードは、システムをネットワークに追加する手順をガイダンスで案内します。このウィザードは、推奨される順序でアプリケーションを示し、状況に応じてどのアプリケーションが適切かを判断するための情報を提供します。NetWork Setup ウィザードについては、『ネットワーク管理ガイド：接続編』および SysMan Menu のオンライン・ヘルプを参照してください。

- リンク・アグリゲーション

システム管理者は、複数の物理 Ethernet ネットワーク・インタフェース・カード (NIC) を結合して、「リンク・アグリゲーション・グループ」を作成することができます。上位層のソフトウェアには、このリンク・アグリゲーション・グループが 1 つの論理インタフェースに見えます。

リンク・アグリゲーション (「トランキング」とも呼ばれます) により、次の機能が利用できます。

- NIC の数とタイプに従った、ネットワーク帯域幅の拡大
- リンク・アグリゲーション・グループのすべてのポートへの負荷分散

リンク・アグリゲーションは、サーバ対サーバ接続、サーバ対スイッチのポイント・ツー・ポイント接続でのみサポートされます。詳細は、3.2.10 項を参照してください。

- イベントの分析と管理

- collect ユーティリティ

collect ユーティリティは、特定のオペレーティング・システム・データの記録や表示を行うシステム監視ツールです。collect を使用すると、ファイル・システム、メッセージ・キュー、tty、またはヘッダなどのサブシステムの単位で、データ収集に含めたり除外することができます。データは端末に表示することも、データ・ファイル (圧縮形式または非圧縮形式) に格納することもできます。データ・ファイルは、コマンド行、コマンド・スクリプト、またはグラフィカル・ユーザ・インタフェースを使って、読み取ったり操作することができます。

- Compaq Analyze

Compaq Analyze を使用すると、システムのイベント・ログ・ファイルに保存されているイベントを分析できます。この GUI は、規則ベースのハードウェア障害管理診断ユーティリティです。ユーザは、プログラム・パラメータと構成パラメータを設定し、イベント情報を調べることができます。Compaq Analyze についての詳細は、2.6.2 項を参照してください。

- Event Manager

システムのコンポーネントがイベントや状態の報告に利用できるように、ログ・ファイルのようなさまざまな場所に分散して存在

する情報を単一のポイントで提供するためのアプリケーションです。詳細は 2.6.7 項を参照してください。

- DECevent

DECevent イベント管理ユーティリティは、エラー報告とバイナリからテキストへの変換の機能を提供します。DECevent によって、さまざまなプラットフォームに対してシステム指定の診断機能を提供します。DECevent についての詳細は、2.6.8 項を参照してください。

- HP Insight Manager によるシステム監視

HP Insight Manager は、Web-Based Enterprise Services (WEBES) を使用して、システムを監視するための Web ベースの環境を提供します。HP Insight Manager は、SysMan およびその他の Tru64 UNIX ユーティリティ用の一貫したラッパを提供します。HP Insight Manager についての詳細は、2.6.1 項を参照してください。

- その他の主要なユーティリティ

次の大半のユーティリティは、Sysman Station、SysMan Menu、コマンド行、および共通デスクトップ環境のアプリケーション・マネージャから利用できます。これらのツールの使用方法については、『システム管理ガイド』と、各リファレンス・ページを参照してください。

- ハードウェア管理 (hwmgr)

hwmgr ユーティリティを使用すると、ディスクやテープ・ドライブ、プロセッサ、バスなどのハードウェア構成要素を容易に管理できます。hwmgr についての詳細は、2.8 節を参照してください。

- 環境モニタリング

envconfig ユーティリティは、AlphaServer システムの温度、ファン、および冗長電源の状態をハードウェア・センサで監視します。環境モニタリングについての詳細は、2.6.6 項を参照してください。

- システム・パラメータの分析 (sys_check)

sys_check ユーティリティは、稼働中のシステムの構成、ハードウェア、およびソフトウェアを出力し、それらの情報を HTML 形式で表示します。また、オペレーティング・システムのパラメータと属性の基本分析も実行し、問題を検出した場合は警告を発します。

- クラス・スケジューラ
クラス・スケジューラを使用すると、システム管理者によるタスクの優先順位付けが可能になります。
- カーネル・チューナ
カーネル・チューナを使用すると、カーネル・サブシステムのパラメータの表示と変更が可能になります。詳細は 5.1.1 項を参照してください。
- プロセス・チューナ
プロセス・チューナは、システム・プロセスを表示、監視、および管理するためのアプリケーションです。複数のソートおよびフィルタ・オプションが提供されるので、情報の表示を管理することができます。詳細は `dxproctuner(8)` を参照してください。
- `spike` によるコードの最適化
`spike` ユーティリティを使用すると、リンク後にコードの最適化を行うことができます。このユーティリティは `om` に代わるものであり、最適化機能は `om` に似ています。`spike` は、プログラム全体を処理対象にできるため、コンパイラでは不可能な最適化を行うことができます。詳細については、`spike(1)` のリファレンス・ページを参照してください。
- Division of Privileges
Division of Privileges ユーティリティを使用すると、管理者は、`root` パスワードなしで特権プログラムにアクセスする権利をユーザやグループに与えることができます。すべての特権システム管理アプリケーションは、SysMan Menu、SysMan Station、デスクトップのアプリケーション・マネージャ、カスタム・セットアップ・チェックリスト、またはコマンド行から起動することができます。詳細は『セキュリティ管理ガイド』を参照してください。
- ブータブル・テープ
ブータブル・テープ機能は、システムからディスク・イメージを作成し、回復するための手段を提供します。

1.2.6 ARMTech リソース管理ソフトウェア

ARMTech リソース管理ソフトウェアを使用すると、ミッション・クリティカルなアプリケーションを最高の性能で稼働させるために必要な高度なリソース管理を簡単に実施することができます。Tru64 UNIX には Aurema 社のリソース管理ソフトウェア製品が含まれています。

Aurema 社のエントリ・レベルのリソース管理ユーティリティである ShareExpress は、ライセンスなしで利用できます。より強化された Aurema リソース管理製品 ShareExtra および ShareEnterprise は、Aurema 社からライセンスを購入することにより使用可能になります。これらの製品のキットは Tru64 UNIX の「Associated Products Volume 2」CD-ROM に含まれています。

ARMTech は、CPU システム・リソースの動的な割り当てとバランシングを可能にする強力なシステム・ユーティリティです。

ライセンス・フリーの ShareExpress 製品には、ユーザごとにリソースを公平に配分する拡張 UNIX タイムシェアリング・スケジューラが用意されています。ShareExtra 製品を使用すると、特定のユーザやアプリケーションに対して CPU リソースの配分を変えることができます。ShareEnterprise 製品は、グループ単位での CPU リソースの分配、履歴情報と課金情報の収集、その他の強力なリソース管理機能を提供します。

1.2.7 UNIX と Microsoft Windows の相互運用性

Tru64 UNIX では、UNIX と Microsoft Windows が混在する環境におけるソリューションの開発、導入、および管理を単純化するための機能と製品を多数用意しています。これらの機能によって、さまざまな機能領域で統合に関する問題の解決が容易になります。詳しい説明は、第 10 章を参照してください。

1.2.8 Advanced Printing Software

Advanced Printing Software は、Tru64 UNIX のプリント・システムです。このソフトウェアは、Xerox 社の PrintXchange 技術をベースに Xerox 社と共同で開発されました。Advanced Printing Software は、ワークグループおよびエンタープライズ環境のための分散クライアント/サーバ・プリント・システムです。

Advanced Printing Software には、ジョブ・スケジューリング、ジョブ保留、イベント通知、複数レベルのアクセス制御、プリント・キューのフェイルオーバー、および TruCluster Server 環境で使用する際のホスト透過性などのプリント・スプーリング機能がすべて用意されています。

このソフトウェアは NIS (Network Information Service) または LDAP (Lightweight Directory Access Protocol) を使用して、ユーザの環境内にプリンタ情報を配布します。また、インバウンド・ゲートウェイとアウトバウンド・ゲートウェイを使用して、lpd プリント・サブシステムと連携して動作します。

Advanced Printing Software は、ISO 10175 ドキュメント・プリンティング・アプリケーション標準で定義されているプリント・システム・モデルと、POSIX 1384.7 草案で定義されているコマンド・セットをベースとしています。

詳細は、『*Advanced Printing Software* リリース・ノート』および『*Advanced Printing Software* ユーザ・ガイド』を参照してください。

1.2.9 System V のサポート

System V 環境は、コマンド、サブルーチン、およびシステム・コールの代替バージョンで構成されており、ベース・システムおよびカーネル拡張のすべての構成要素に関して、System V インタフェース定義 (SVID) で定義されているソース・コード・インタフェースと実行時動作をサポートしています。System V 環境を使用すると、省略時のシステム・コマンドとシステム関数の代わりに、System V 対応のコマンドおよび関数 (システム・コールとサブルーチン) が使用されます。Tru64 UNIX System V 環境では、SVID2 と SVID3 のすべての機能がサポートされています。

System V 環境については、『*Tru64 UNIX* ユーザーズ・ガイド』を参照してください。

1.2.10 Secure Web Server

Secure Web Server は、業界標準の Apache Software Foundation (ASF) のコードをベースとしています。HP では、156 ビットまでの暗号化が可能な SSL (Secure Sockets Layer) 機能を含めることにより、ベースとなる ASF 製品を改良しています。Secure Web Server には、Java Servlets、Java Server Pages、および PHP Hypertext Preprocessor のサポートも含まれています。

1.2.11 Tru64 UNIX のフリーウェア

Tru64 UNIX のメディア・キットには、フリーウェア CD-ROM が含まれています。この CD-ROM には、Tru64 UNIX システムの性能を利用したマルチメディア・パッケージとグラフィック・パッケージとともに、ユーザ・ユーティリティおよび管理ユーティリティが収められています。このフリーウェア集を使用すると、HP Tru64 UNIX での動作が確認されたソフトウェアをインターネット上で探す手間を省略することができます。

フリーウェア CD-ROM には、パブリック・ドメイン・フリーウェアおよびシェアウェア・プログラム集の他に、HP が開発したツール (非商用ツールおよび商用パッケージのデモ版) も収められています。Internet Express for Tru64 UNIX CD-ROM について説明した 1.3.7.1 項も参照してください。

1.2.12 ドキュメント

Tru64 UNIX のドキュメントによって、必要な情報をより使いやすい形式で入手できます。V5.1B での変更点も含めた Tru64 UNIX のドキュメントに関するすべての情報については、『ドキュメント概要』を参照してください。以下に、Tru64 UNIX のドキュメントの機能の一部を説明します。

- ドキュメントの大部分は、Web ブラウザで見るための HTML 形式、Adobe Acrobat Reader で見たり印刷したりするための PDF 形式、および印刷された書籍の形式で利用できます。

ブラウザの印刷オプションを使用すると、必要なセクションを簡単に印刷できます。また、内容をカット・アンド・ペーストして、特定のタスクを実行するための独自のドキュメントを作成することができます。Acrobat Reader を使うと、活字で組んだものとほとんど同等の品質で、節、章、またはドキュメント全体を印刷できます。

- Tru64 UNIX のドキュメント CD-ROM は、Tru64 UNIX システムだけでなく、Windows PC または Macintosh のラップトップまたはワークステーションでも使用できます。また、ISO 9660 レベル 1 の CD-ROM 規格に準拠したシステムならどれでも利用できます。また、英語版ドキュメント CD-ROM には、各オペレーティング・システム用の Adobe Acrobat Reader が含まれています。

ドキュメント CD-ROM に含まれているマニュアルの中から特定の情報を検索するための機能として、日本語ドキュメント CD-ROM には日本

語全文検索機能 midoc が含まれています。詳細は『日本語機能ガイドブック』を参照してください。

英語版のドキュメント CD-ROM には、HTML 文書のマスター・インデックスとともに AltaVista Search CD-ROM ソフトウェアが入っています。この AltaVista ソフトウェアは、Windows 95 以降の Windows オペレーティング・システムが稼働する x86 ベースのパソコンで実行できます。

- 印刷されたハードコピー・ドキュメントは、オプションで注文できるドキュメンテーション・キットとして用意されています。詳細は、『ドキュメント概要』を参照してください。
- Tru64 UNIX のドキュメントは、以下の Web サイトで参照することもできます。

<http://tru64unix.compaq.co.jp/document/index.html>

また、この Web サイトでは、Tru64 UNIX のリファレンス・ページや次のドキュメントにもアクセスできます。

- Tru64 UNIX Best Practices ドキュメント
- Tru64 UNIX Device Driver マニュアル・ライブラリ (英語)
- TruCluster Server のドキュメント
- Advanced Server for UNIX のドキュメント
- Internet Express for Tru64 UNIX (1.3.7.1 項を参照) のドキュメント

英語版のドキュメント CD-ROM には、*isearch* という英文ドキュメント用の全文検索ツールが含まれています。この検索ツールを使用すると、任意のプラットフォーム (UNIX ワークステーション、Macintosh コンピュータ、Windows PC など) で、Web 上やローカル・エリア・ネットワーク上の Tru64 UNIX ドキュメントの検索ができます。

- Web 上の Tru64 UNIX ドキュメントにアクセスする場合は、検索ツールに単語を入力するだけで、ドキュメント内でそのトピックについて説明している場所を見つけることができます。
- LAN 上で検索ツールを使用するには、システム管理者がローカル・サーバ上に *isearch* ソフトウェアをインストールしていなければなりません。インストール後、そのサーバにアクセスしているユーザは、Web の場合と同様の方法でドキュメントを検索できます。英語

版のドキュメント CD-ROM には、isearch データベース、検索エンジン、設定方法を説明したドキュメントが含まれています。

- Best Practice ドキュメントでは、それぞれの機能で利用可能なすべてのオプションを説明する代りに、タスク実行のための推奨される方法を要約して提供しています。これらのドキュメントでは、実行手順を段階的に説明しています。
- SysMan および関連するアプリケーションのオンライン・ヘルプは、強力かつタスク指向なので、一定の目的のために必要な情報を容易に検索できます。

1.3 オプション・コンポーネント

Tru64 UNIX オペレーティング・システムで実行できるさまざまなソフトウェア製品が、HP および他社から購入できます。

以下の項で説明する、別ライセンスの製品の大部分は、Tru64 UNIX のメディア・キットに収められています。これらのオプション・ソフトウェア・パッケージについての詳細は、『*Tru64 UNIX Operating System QuickSpecs*』か、Tru64 UNIX の Web サイトを参照してください。

1.3.1 TruCluster Server

TruCluster Server は、Tru64 UNIX ソフトウェア、複数の AlphaServer システム、および多数のストレージを高度に統合し、単一のシステムとして稼働させるためのソフトウェアです。TruCluster Server クラスタは、複数のシステムから構成されていても、単一の仮想システムとして動作します。クラスタのメンバは、単一のセキュリティと管理ドメインで、リソース、データ・ストレージ、クラスタ単位のファイル・システムを共有する一方、各メンバは、クラスタを中断せずに独立してブートまたはシャットダウンできます。

TruCluster Server 環境は、必要に応じて単純化または多機能化することができます。必要に応じて、2 ノードから 8 ノードまでのクラスタを構成し、次のような可用性の高いアプリケーションを実行できます。

- トランザクション処理システム
- ネットワーク・クライアント/サーバ・アプリケーション用のサーバ
- ダウンタイムを最小限に抑えなければならないデータ共用アプリケーション

- TruCluster Server のアプリケーション・プログラミング・インタフェース (API) を最大限に利用した分散並行処理アプリケーション

TruCluster Server には、インターネット・プロトコル・セット (TCP/IP) のためのクラスタ別名が含まれているので、ネットワーク・クライアントやピア・システムはクラスタを単一のシステムとして認識します。

TruCluster Server は単一システムの管理機能をクラスタに拡大したもので、Tru64 UNIX システムの管理方法を理解していれば、TruCluster Server クラスタの管理方法もすでに理解していることになります。SysMan Menu ユーティリティはこのクラスタ環境全体を統合して表示するので、単一のメンバやクラスタ全体を管理することができます。

詳細については、『クラスタ概要』を参照してください。

1.3.2 Logical Storage Manager

Logical Storage Manager (LSM) は、オプション機能として提供される、統合化された、ホスト・ベースのディスク・ストレージ管理アプリケーションです。この機能により、ストレージ・デバイス上のデータにアクセスするユーザやアプリケーションに影響を与えることなく、ストレージ・デバイスを管理できます。LSM が提供する機能には、連結、ストライピング、ミラーリング、および RAID 5 があります。LSM の管理作業は、グラフィカル・インタフェース、メニュー方式のユーティリティ、または LSM コマンド行を使って実行できます。

LSM グラフィカル・インタフェース `lsmsa` は、LSM オブジェクト、AdvFS ドメイン、およびそれらの関係を階層構造で表示します。`lsmsa` を使用すると、ローカル・システムまたはリモート (クライアント) システム上の LSM オブジェクトおよび AdvFS ドメインを表示して管理できます。

LSM についての詳細は、2.4 節 および『*Logical Storage Manager*』を参照してください。

1.3.3 Advanced File System Utilities

Advanced File System Utilities は、AdvFS の可用性と柔軟性を向上させます。このユーティリティを使用すると、次のような効果が期待できます。

- ファイル・システム管理時間の簡略化
- データ・オンライン状態での日常保守

- ファイル容量の拡大
- ボリューム間での使用率の平均化
- ホット・バックアップのためのファイルのリストア、ストライプ、クローン作成

1.3.4 Developer's Toolkit

Developer's Toolkit は、HP Tru64 UNIX のすべての開発用のツール、言語、および環境の前提条件となるものです。この Toolkit には、次のコンポーネントが含まれています。

- ANSI C 準拠の HP C for Tru64 UNIX
- Ladebug および dbx デバッガ
- Atom API
- プログラム分析ツール (プロファイリングと性能分析を行います)
- Visual Threads (スレッド関連のプロファイリングとデバッグを行います)
- Porting Assistant

詳しい説明は、第 6 章を参照してください。

1.3.5 Advanced Server for UNIX

Advanced Server for UNIX は、Tru64 UNIX サーバ、Windows NT サーバ、Windows 2000 サーバ、および Microsoft Windows クライアントの間でシームレスな相互運用性を実現します。Advanced Server によって Tru64 UNIX システムは、Microsoft クライアントに対して Microsoft Advanced Server と同じようにサービスを実行することができます。詳しい説明は、10.1 節を参照してください。

1.3.6 Multimedia Services

Multimedia Services ソフトウェアによって、HP Tru64 UNIX ワークステーションにオーディオおよびビデオ機能が提供されます。また、完全なマルチメディア・プログラミング・ライブラリを開発者に提供します。Multimedia Services Run-time ライセンスは、基本オペレーティング・システムに含まれています。

1.3.7 その他のソフトウェア

以降の項では、Tru64 UNIX メディア・キットに含まれていないソフトウェアについて説明します。これらのソフトウェアは、必要に応じて別途購入できます。

1.3.7.1 Internet Express for Tru64 UNIX

Internet Express for Tru64 UNIX CD-ROM には、Tru64 UNIX AlphaServer システム上で動作するようにコンパイルされた、一般的に使用されるインターネット・ソフトウェアが収録されています。Internet Express の CD-ROM は AlphaServer システムに付属しています。また、HP に別途注文することもできます。

このソフトウェア・パッケージには、HP が開発した、インターネット・コンポーネントを構成/管理するための管理ツールも含まれています。このパッケージに含まれているソフトウェアの一部を、以下に示します。

- オープン・ソース・インターネット・ソフトウェア集 (バイナリとソース)
Tru64 UNIX サーバ (Web、メール、ニュース、ディレクトリ、プロキシ、およびストリーミング・メディア)、データベース管理システム、セキュリティ・サービス、およびダイナミック Web コンテント生成用ツール上で動作が確認されています。
- オープン・ソース・インターネット・ソフトウェアの自動インストールおよび構成
- 販売用インターネット製品集 (完全機能版または評価版)
- Secure Web Server (Tru64 UNIX のメディア・キットにも含まれています。 1.2.10 項を参照してください。)
- Internet Monitor (Web ベースのサービス品質管理ツール)
- インターネット・サービスを管理する、Web ベースの管理ユーティリティ

すべてのソフトウェアは、TruCluster Server 上での動作がテストされています。Internet Express for Tru64 UNIX についての詳細は、次の Web サイトを参照してください。

<http://tru64unix.compaq.co.jp/products/internet/ix.htm>

Tru64 UNIX のフリーウェア CD-ROM について説明した 1.2.11 項も参照してください。

1.3.7.2 Enterprise Toolkit for Visual Studio

この Enterprise Toolkit for Visual Studio は、Tru64 UNIX サーバ用の C、C++、および Fortran アプリケーションの開発をサポートする Microsoft Visual Studio の拡張機能またはアドインのセットです。Enterprise Toolkit for Visual Studio を使用すると、開発者は、広く普及している Microsoft Visual Studio ツールを使用して、Tru64 UNIX または Windows 用のアプリケーションの開発、編集、コンパイル、構築、およびデバッグを単一のデスクトップから実行できます。

単一のパソコン・ツール群を使用して、64 ビット Tru64 UNIX Alpha テクノロジを利用しながら、基本的な UNIX アプリケーションの作成および管理や、より強力で複雑なクライアント・サーバ型の分散アプリケーションの作成が可能です。

さらに、Enterprise Toolkit によって、開発者は、性能とメモリを分析するための多数のツールを利用できます。Enterprise Toolkit は、Visual Studio のマニュアル・ブラウザである HTML Help Viewer を使用して、開発者が Windows のマニュアルを読むのと同じウィンドウから UNIX および製品のマニュアルへアクセスできるようにします。

1.3.7.3 ソース・マテリアル・オプション

ソース・モジュールの検索と修正が必要なユーザ向けに提供されるソース・キットです。主に、非常に特殊な修正をする場合に利用されます。

1.4 パッケージの内容

Tru64 UNIX メディア・キットには、次の CD-ROM が入っています。

- Operating System
- Associated Products Volume 1
- Associated Products Volume 2
- Software Documentation
- Alpha Systems Firmware Update
- フリーウェア

- Linux and HP Tru64 UNIX Portability Tools

- 日本語追加機能

関連製品の多くは、基本オペレーティング・システムのオプション・サブセットとなっています。Developer's Toolkit、TruCluster Server、Logical Storage Manager、および Advanced Server for UNIX などの製品は、オペレーティング・システムのライセンスとは別に個々の製品のライセンスが必要です。詳しい説明は、1.3.7 項を参照してください。

インストール・キットに含まれる CD-ROM およびドキュメントの詳細は、カスタマー・レター『Tru64 UNIX をご使用のお客様へ』を参照してください。

1.4.1 マニュアル

次の印刷マニュアルが Tru64 UNIX メディア・キットに同梱されています。

- 『リリース・ノート』
- 『インストール・ガイド』
- 『インストール・ガイド — 上級ユーザ編』
- 『*Update Installation Quick Reference Card*』
- 『*Full Installation Quick Start*』カード
- 『Tru64 UNIX 概要』
- 『クラスタ概要』
- 『*Quick Reference Card*』
- 『ドキュメント概要』
- 『日本語機能ガイドブック』

1.4.2 ライセンス

Alpha システムでは、他のすべてのライセンスの前提条件であるオペレーティング・システムのベース・ライセンスに加えて、次の 4 種類のオペレーティング・システム・ライセンスが利用できます。

- SMP 拡張ライセンス
- コンカレント・ユース・ライセンス

- 無制限ユーザ・ライセンス
- ハードウェア・パーティショニング・ライセンス

Tru64 UNIX は、特定のサーバでスタティック・ハードウェア・パーティションをサポートするためのテクノロジーを提供しています。ハードウェア・パーティション機能により、複数のオペレーティング・システムを使用することが可能になり、新規バージョンのテストやアプリケーションの複数のバージョンの実行における柔軟性が向上します。ハードウェア・パーティションで Tru64 UNIX を使用するには、パーティションごとに Tru64 UNIX ハードウェア・パーティショニング・ライセンスが必要です。ハードウェア・パーティショニングについての詳細は、『システム管理ガイド』を参照してください。

これらのライセンスについての詳細は、『*Tru64 UNIX Operating System QuickSpecs*』（以前の『ソフトウェア仕様書』）を参照してください。

システム管理

この章では、Tru64 UNIX オペレーティング・システムのシステム管理機能の一部を紹介します。次のトピックについて説明します。

- オペレーティング・システムとオプション・サブセットのインストール (2.1 節)
- オペレーティング・システムおよび TruCluster Server のパッチ・キットのインストール (2.2 節)
- システムの構成 (2.3 節)
- Logical Storage Manager を使用したストレージ・デバイスの管理 (2.4 節)
- グラフィカル・アプリケーション SysMan Menu によるシステム管理 (2.5 節)
- システムとイベントの監視および管理 (2.6 節)
- クラッシュ・ダンプの管理 (2.7 節)
- hwmgr ユーティリティと DDR (Dynamic Device Recognition) フレームワークを使用したハードウェア管理 (2.8 節)
- サブシステムの動的なロード (2.9 節)
- カーネル属性の動的な変更 (2.10 節)
- データレス管理サービス (2.11 節)

2.1 インストール

Tru64 UNIX は、フル・インストールとアップデート・インストールの両方を、CD-ROM から、またはネットワーク経由で RIS (Remote Installation Services) サーバからサポートします。また、Tru64 UNIX は、システムのクローニングもサポートしています。

2.1.1 フル・インストールレーション

フル・インストールレーションによって、Tru64 UNIX を新規にあるいは既存のシステムにインストールできます。ファイル・システムのレイアウト、カーネルの構成要素、およびインストールするソフトウェア・サブセットに関して推奨される設定を使用することができます。また、独自のカスタマイズを行うこともできます。

オペレーティング・システムは、グラフィカル・インタフェースあるいはテキスト・ベースのインタフェースのいずれかを使用して、素早く簡単にインストールすることができます。グラフィックス機能を備えたシステムで利用できる GUI では、設定プロセスの各ステップを自由に移動できます。テキスト・ベースのインタフェースの場合も同様に、設定作業の各ステップが表示され、必要に応じて前の段階に戻って設定を変更することができます。

フル・インストールレーションでは、ファイル・システムとスワップ領域を新規に作成するため、インストール対象となるディスク・パーティション上の既存のシステムあるいはユーザ作成ファイルは上書きされます。ディスク・レイアウトおよびスワップ領域の割り当てには、省略時の値を使用することも、設定をカスタマイズすることも可能です。

フル・インストールレーションの主な機能は、次のとおりです。

- テキスト・ベースのインタフェースも GUI も、タスク指向で設計されています。これにより、インストール・プロセスの各ステップごとに処理のためのメニューが表示され、いつでも前後のステップに移動して選択した内容を変更できるようになっています。
- フル・インストールレーション・プロセスは、選択したソフトウェアにもとづいて、ファイル・システムのレイアウトを決定します。ファイル・システムのサイズを計算したり、正常なインストールのためにインストールの前にディスクのパーティションを再設定する必要はありません。
- インストール後の作業としてではなく、フル・インストールレーション中に、Logical Storage Manager (LSM) と ワールドワイド言語サポート (WLS) をインストールし、構成することができます。

カスタム・スクリプトまたはプログラムを作成し、インストールレーション中に 3 つの処理ポイント、すなわち、インストールレーションの開始前、ソフトウェア・サブセットをロードした後、およびシステム・リブート後のいずれかで実行することにより、フル・インストールレーションをカスタマイズまたは拡

張することができます。作成したファイルは、インストール・プロセスで使用するために、ディスク、CD-ROM、または RIS サーバのいずれかにロードします。

フル・インストールにより、`install.cdf` という構成記述ファイル (CDF) が作成されるので、これを使用して、同様のシステムでフル・インストールを複製することができます (2.1.3 項参照)。また、実行中のシステムから構成データを `config.cdf` ファイルに取り出し、フル・インストール時に次の情報を複製することができます。

- ネットワーク・カードおよびルータの構成
- DNS (Domain Name Service)
- NIS (Network Information Service)
- NTP (Network Time Protocol)
- プリンタ・サービス
- メール・サービス

フル・インストールの詳しい説明は、『インストール・ガイド』および『*Full Installation Quick Start*』カードを参照してください。

2.1.2 アップデート・インストール

アップデート・インストールは、Tru64 UNIX を、オペレーティング・システムの以前のバージョンから更新します。Version 5.1B のアップデート・インストール・プロセスを使用して、システムを Version 5.1 あるいは 5.1A から Version 5.1B へアップデートできます。アップデート・インストールでは、次の設定は維持されます。

- ディスク・パーティション
- ファイル・システム
- ファイルのカスタマイズ
- ネットワーク環境、プリント環境、およびメール環境
- ユーザ・アカウントとユーザ作成ファイル
- システムに対するその他のカスタマイズ

ファイル・システムのタイプ、記憶位置、またはサイズを変更する場合や、オプションのソフトウェアをインストールする場合は、アップデート・インストールは実行しないでください。

アップデート・インストールでは、次のソフトウェア・サブセットが更新されます。

- オペレーティング・システムを構成するソフトウェア・サブセット (基本ソフトウェア・サブセットと呼ぶ) で現在インストールされているもの。基本ソフトウェア・サブセットのサブセット名の先頭には、OSF という文字が付きます。
- 新しいバージョンで導入された必須の基本ソフトウェア・サブセット
- 現在インストールされている ワールドワイド言語サポート (WLS) ソフトウェア・サブセット。WLS ソフトウェア・サブセットのサブセット名の先頭には、IOS という文字が付きます。
- 現在インストールされている TruCluster ソフトウェア・サブセット。TruCluster ソフトウェア・サブセットの名前には、TCR というプレフィックスが付きます。

-u オプションを使用して起動すると、アップデート・インストール・プロセスは自動モードで実行されます。自動モードでは、アップデートに関して特に問題がなければ、ユーザの介入なしでインストールが進められます。唯一の例外は、WLS ソフトウェアをインストールする際の CD-ROM の取り替えです。-u オプションでは、すべてのカーネル・コンポーネントを組み込んでカーネルが構築され、また、古いファイルの保管は行われません。

アップデート・インストール・プロセスの冒頭の分析フェーズでは、次の処理が行われます。

- アップデート・インストールに支障をきたすレイヤード・プロダクトの存在を検出します。
- アップデート後に再インストールの必要があるレイヤード・プロダクトを検出します。
- ファイル・システムのタイプの不一致を発見します。
- 利用可能なディスク領域を調べます。

レイヤード・プロダクトまたは回復可能なファイル・タイプに関する矛盾が見つかった場合、それらをアップデート・インストールのユーザ・イ

ンタフェースから直接解決できます。 インストレーション・プロセスを終了させてから矛盾を解決し、インストールを再実行する必要はありません。 新しいソフトウェアのためのディスク領域やアップデート・インストレーションのための領域が不足している場合、ディスク領域回復オプションを、アップデート・インストレーションから直接利用できます。

アップデート・インストレーション時には、オプションのソフトウェア・サブセットを追加インストールしたり、レイヤード・プロダクトをアップデートしたりすることはできません。 この場合、アップデート・インストレーション完了後に、setldユーティリティを使用して、オプションのソフトウェア・サブセットを追加でインストールします (2.1.6 項 参照)。 レイヤード・プロダクトをアップデートするには、既存のバージョンを削除して、新しいバージョンのオペレーティング・システムに対応した新しいバージョンをインストールしなければならない場合もあります。 その必要がある場合、アップデート・インストレーション・プロセスによって、そのことがユーザに通知されます。

アップデート・インストレーションの機能には 2 種類あります。

- アップデート・インストレーションの開始時に、ユーザが制御する機能 (表 2-1)。
- アップデート・インストレーション・プロセスに組み込まれている機能 (表 2-2)。

表 2-1: ユーザが制御するアップデート・プロセスの機能

ユーザ・オプション	説明
自動アップデート・インストレーション	オプションのカーネル・コンポーネントを選択する必要がない場合、または古いファイルをアーカイブする必要がない場合、-u フラグを付けてアップデート・インストレーションを起動することにより、ユーザの介入なしにアップデートを実行できる。
カーネル・コンポーネントのオプション	必須カーネル・コンポーネントのみをカーネルに組み込むか、すべてのカーネル・コンポーネントをカーネルに組み込むか、またはオプションのカーネル・コンポーネントを対話的に選択するかが選べる。
古いファイルのアーカイブ	古いファイルがアップデート・インストレーション・プロセスによって自動的に削除される前に、それらをアーカイブするオプションが用意されている。

表 2-2: アップデート・プロセスの内蔵機能

組み込み機能	説明
競合するレイヤード・プロダクトの検知	インストールされたレイヤード・プロダクトが、新しいバージョンのオペレーティング・システムと互換性がない可能性がある場合、それを通知する。そのレイヤード・プロダクトは、後で再インストールしなければならないことがある。
アップデートの続行を妨げるレイヤード・プロダクトの削除	ユーザの確認後、アップデートの続行を妨げるレイヤード・プロダクトが削除される。
ベース・オペレーティング・システムおよび WLS ソフトウェアの新バージョンへの更新	インストールされている既存のファイルを更新し、新しいバージョンで新規に導入された必須サブセットをインストールする。
ファイル・タイプの変更のチェック	ファイル・タイプが変更されているかどうかをチェックする。矛盾が発生した場合には、アップデートを実行できないことがある。
ディスク容量の回復	アップデートを完了するのに必要なファイル容量が足りない場合、不必要なサブセットと <code>.PreUPD</code> 、 <code>core</code> 、および余分のカーネル・ファイルを削除して、ディスク容量を回復するオプションを提供する。
ユーザ提供ファイル内の命令の実行	ユーザ提供スクリプト、プログラム、または実行可能ファイルを適切な位置に作成あるいは移動することにより、アップデート・インストールーションをカスタマイズすることができる。アップデート・プロセスが、所定の位置で正しい名前のファイルを見つけた場合、そのファイルが実行される。

アップデート・インストールーションの詳細については、『インストールーション・ガイド』および『*Update Installation Quick Reference Card*』を参照してください。

2.1.3 インストールーションのクローニング

インストールーションのクローニングによって、稼働中のシステムから、同一または類似のハードウェア構成の 1 つ以上のシステムへ、インストール属性、つまりファイル・システムとインストールされたソフトウェアを複製できます。

多数のシステムにインストールーションを行う場合、インストールーションのクローニングを使用すると次の利点があります。

- インストールがより簡単に実行できます。

- ユーザの最小限の介入で実行できるように、インストールのクローニングのためのプロセスを設定できます。
- インストールのクローニングによって、各システムに対して手動でインストールを実行しなくてすむため、時間が節約でき、環境にエラーが生じる可能性が減ります。
- CD-ROM のようなローカルにマウントされる着脱可能なメディアを使用して並行インストールする場合と違い、ソフトウェアを集中管理できます。

オペレーティング・システムの最新バージョンをマシンにインストールする場合、ユーザが指定したインストール設定データを含む構成記述ファイル (CDF) がインストール・プロセスによって自動的に生成されます。したがって、対象とするシステムに同一のインストールを実行するために必要なインストール情報は、すべてこの CDF に入っています。

あるバージョンのオペレーティング・システムが作成した CDF は、別のバージョンのオペレーティング・システムとは互換性がないので、バージョンの異なるオペレーティング・システム間ではインストールのクローニングはサポートされません。したがって、対象とするシステムに Version 5.1B のクローンを作成する場合は、Version 5.1B のフル・インストールを実行して CDF を作成しなければなりません。

クローニングを実行するシステムは、CDF が生成されたシステムと同一のディスク構成になっている必要があります。つまり、/(ルート)、/usr、/var、/usr/i18n ファイル・システムとスワップ領域について両方のシステムで使用されるディスクには、同一のディスク・タイプとデバイス名が必要です。ただし、構成上のわずかな違いは調整できます。

インストールのクローニングの詳しい説明は、『インストール・ガイド — 上級ユーザ編』を参照してください。

2.1.4 構成のクローニング

構成のクローニングによって、すでに構成されているシステムから 1 つ以上のシステムへ、ネットワーク、プリンタ、メール・サービスなどの設定項目を複製できます。構成のクローニングが実際に役に立つのは、2 つ以上のシステムを類似の形式で構成する場合です。

新たなシステムのインストレーションと構成を完全に自動化するため、構成のクローニングとインストレーションのクローニングを併用すれば、システムのインストール後に構成タスクを手動で実行しなくてすむようになっています。

構成のクローニングは、バージョンの異なるオペレーティング・システム間ではサポートされません。

詳細は『インストレーション・ガイド』および『インストレーション・ガイド — 上級ユーザ編』を参照してください。

2.1.5 クラスタのローリング・アップグレード

ローリング・アップグレードとは、クラスタ・システムのソフトウェアのアップグレードをクラスタの動作中に行うものです。クラスタ・メンバ・システムを1台ずつアップグレードして、アップデートが完了したら動作状態に戻す一方で、クラスタ環境内で複数のバージョンの基本オペレーティング・システム、クラスタ・ソフトウェア、およびワールドワイド言語サポート (WLS) ソフトウェアが存在する状態を自動的に管理します。サービスにアクセスしているクライアントには、ローリング・アップグレードが進行中であることは意識させません。

ローリング・アップグレードは、クラスタ環境で新しいリリースのオペレーティング・システムおよびクラスタ・ソフトウェアをインストールする唯一の方法です。また、クラスタ環境でパッチ・キットをインストールする場合にも使用できます。クラスタ環境でパッチを適用する方法については2.2節 および「*Patch Kit Installation Instructions*」を参照してください。ローリング・アップデートについての詳細は『クラスタ・インストレーション・ガイド』を参照してください。

2.1.6 setld ユーティリティ

setld ユーティリティを使用すると、システム管理者は、『*Guide to Preparing Product Kits*』で示すガイドラインに従ってフォーマットされたソフトウェア・サブセットのインストール、インストールしたサブセットの一覧表示、およびサブセットの削除が可能です。たとえば、システム管理者は、フル・インストレーションまたはアップデート・インストレーションでインストールされなかったオプション・サブセットを setld ユーティリティを使用してインストールすることができます。

アプリケーション・プログラマは、Tru64 UNIX システムにインストールするソフトウェア・サブセットをパッケージングする場合、HP のキットिंग・プロセスを使用する必要があります。

setld ユーティリティの詳細については、『インストール・ガイド』および setld(8) リファレンス・ページを参照してください。

2.1.7 インストール関連のドキュメント

インストール時に役立つドキュメントには、次のものがあります。

- 『インストール・ガイド』
- 『インストール・ガイド — 上級ユーザ編』
- 『*Update Installation Quick Reference Card*』
- 『*Full Installation Quick Start Card*』
- TruCluster Server の『クラスタ・インストール・ガイド』
- 『*Sharing Software on a Local Area Network*』

このドキュメントでは、RIS (Remote Installation Services) と DMS (Dataless Management Services) の設定および使用方法について説明しています。

これらのドキュメントについての情報は、『ドキュメント概要』を参照してください。

2.2 システム・パッチの適用

パッチ・キットは、Tru64 UNIX オペレーティング・システムおよび TruCluster Server ソフトウェアの問題点を修正するためのパッチが含まれたキットです。パッチ・キットのインストールには、dupatch と呼ばれるユーティリティを使用します。このユーティリティは、インストール指示書およびリリース・ノートとともにパッチ・キットに含まれています。

HP は、Tru64 UNIX および TruCluster Server ソフトウェア製品に対しては、リリース・パッチ・キット、CSP (Customer-Specific Patch) キット、ERP (Early Release Patch) キットの 3 種類のパッチを提供します。

- リリース・パッチ

このパッチは、重大な既知の問題の発生を防ぐために提供されるパッチです。リリース・パッチ (正式パッチと呼ぶ場合もある) は、ソフトウェア

ア製品ごとキットにまとめて提供します。このキットはインターネットで提供され、HP に CD-ROM を注文することもできます。

- CSP (Customer-Specific Patche)

このパッチは、特定のハードウェアおよびソフトウェア構成で発生する Tru64 UNIX あるいは TruCluster Server ソフトウェアの問題を解決するためのパッチです。このパッチは、サービス担当者から提供されます。

- ERP (Early Release Patche)

ESP は、次のリリース・パッチ・キットに入れられる予定になっているパッチです。このパッチは、サービス担当者およびインターネットで提供されます。

クラスタ・システムでは、パッチ・キットをインストールする方法が 2 つあります。

- ローリング・アップグレード

この処理は、クラスタが稼働している状態で実行することができます。メンバ・システムを 1 つずつ処理していくため、そのクラスタでは一時的に複数のバージョンのオペレーティング・システム、クラスタ・ソフトウェア、および WLS (Worldwide Language Support) ソフトウェアが存在することになります。ローリング・アップグレードが行われている最中、サービスを受けているクライアントが特に気付くことはありません。

ローリング・アップグレードを行う場合、システムへのパッチの適用には、オペレーティング・システムや TruCluster のインストールと同じプロシージャが使用されます。唯一違う点は、インストール時に `installupdate` ユーティリティを使用する代わりに `dupatch` ユーティリティを使用する点です。

- ローリング・アップグレードを使わない方法

この方法では、1 回の操作でクラスタにパッチを適用しますが、クラスタ全体のリブートが 1 回 (場合によっては 2 回) 必要になります。この方法は、ミッション・クリティカルな環境でなるべく短時間のダウンタイムで素早くパッチを適用したい場合に使用されます。

ローリング・アップグレードを使わない方法でパッチを適用する場合は、プロシージャのいくつかの段階でクラスタが状態を変更し自動的

にリブートを行うので、そのクラスタ上では他の重要な操作は実行すべきではありません。

パッチ・キットは下記の Web サイトからダウンロードできます。

<http://www.hp.com/jp/support/>

パッチ・キットの適用には以下のドキュメントが参考になります。

- *Patch Kit Installation Instructions*

パッチの適用に必要な情報が記述されています。このドキュメントは、それぞれのリリース・パッチ・キット、CSP キット、および ERP キットに含まれています。また、Web サイトから入手することもできます。

- *Patch Summary and Release Notes*

このドキュメントは各リリース・パッチ・キットに含まれています。パッチの適用により変更される点が記述されており、個々のパッチに関してインストール前後に知っておくべき情報が含まれています。すでにリリースされているリリース・パッチ・キットの『*Patch Summary and Release Notes*』は、Web サイトから入手することができます。

- *Patch Kit Roadmap*

すでにリリース済みのリリース・パッチ・キットのキット番号、ベース・レベル、リリース月の一覧が含まれています。

- *Best Practice for Patching Tru64 UNIX*

Tru64 UNIX のリリース・パッチ・キットをインストールする前に検討すべき問題点と、インストールのためのシステムの準備について説明しています。

これらのドキュメントは下記の Web サイトで公開されています。

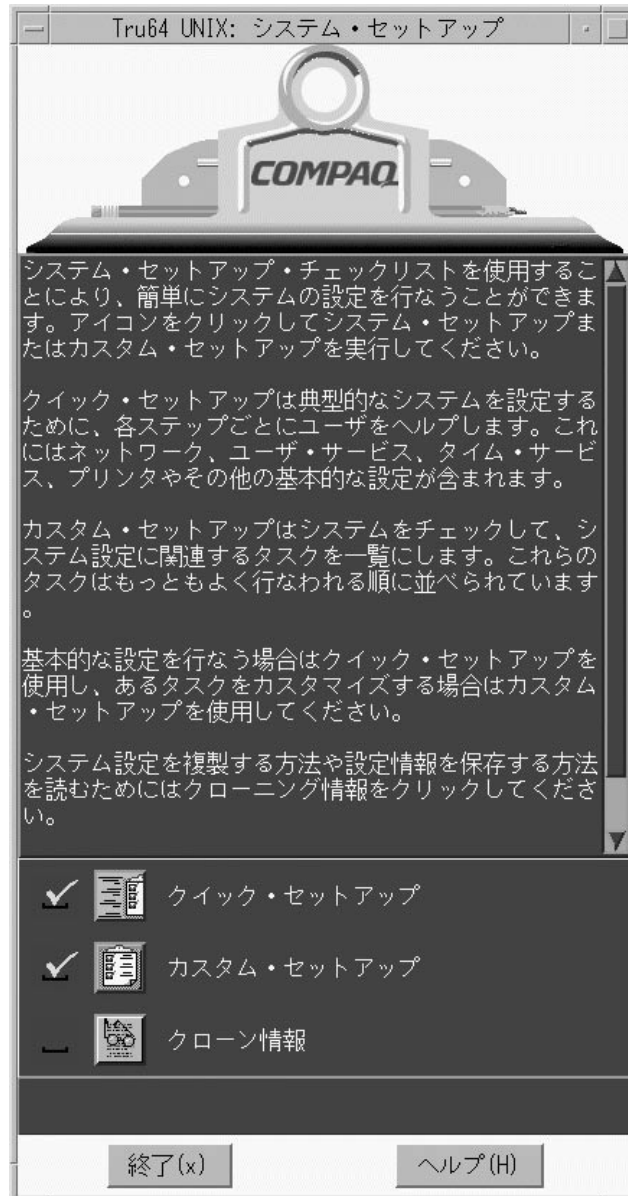
<http://www.tru64unix.compaq.com/docs/patch/>

2.3 システムの設定

グラフィックス機能を備えたシステムの場合、Tru64 UNIX ソフトウェアのインストール後に、SysMan システム・セットアップ・アプリケーション (図 2-1) を使用してシステムの設定を行うことができます。システム・セットアップによって、クイック・セットアップとカスタム・セットアップ・アプリケーションを起動できます。

システムのインストレーション後に初めてスーパーユーザまたは `root` としてログインしたとき、または FIS (factory installed software) システムに初めてログインしたとき、クイック・セットアップを使用して限られたシステム・パラメータ (ネットワークおよびプリンタ・パラメータなど) のみを構成するか、またはカスタム・セットアップを使用して日常業務に使用できるようシステムを詳細設定するかを選択することができます。

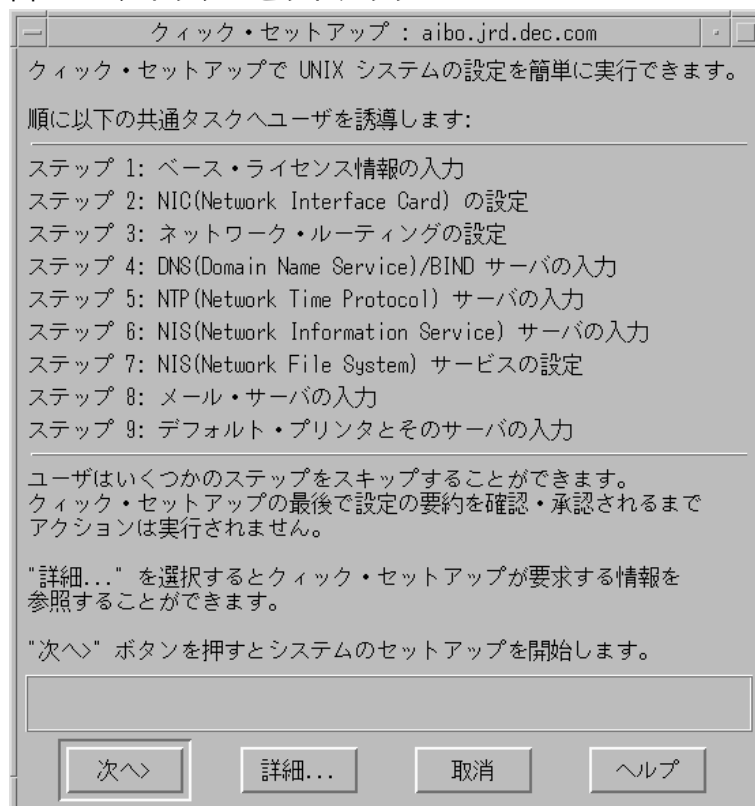
図 2-1: システム・セットアップ



クイック・セットアップ(図 2-2)はウィザード形式で設計されており、システムで最低限必要な情報を入力することができます。クイック・セットアップは、ネットワーク接続、メール、およびプリント機能などの基本的な構成処理を実行します。クイック・セットアップは、ほとんどのシステムに必要な項目をカバーしています。サーバとして構成されるシステムの場合、まず

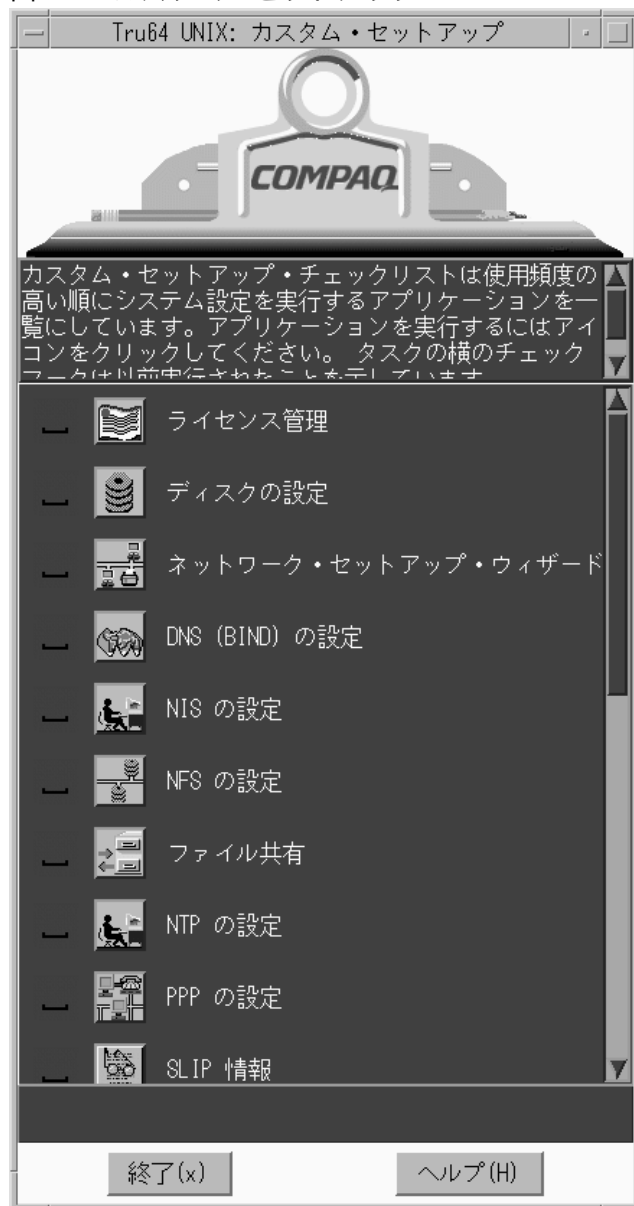
クイック・セットアップを使用してシステムを構成した後、カスタム・セットアップを使用してより詳細な設定を行なってください。

図 2-2: クイック・セットアップ



カスタム・セットアップ (図 2-3) は、クイック・セットアップ以上に高度なシステム構成を行います。カスタム・セットアップを実行すると、まずシステムがチェックされ、システムに適用できる構成アプリケーションのリストが表示されます。これらのアプリケーションは、チェックリストとして表示されます。一度アプリケーションにアクセスすると、アプリケーション名の横にチェックマークが表示されます。個々のアプリケーションの詳細については、「SysMan の紹介」オンライン・ヘルプを参照してください。

図 2-3: カスタム・セットアップ



スーパーユーザまたは root としてログインすると、「システム管理」フォルダの「システム・セットアップ・チェック」アイコンをクリックすることにより、またはコマンド行に次のコマンドを入力することにより、SysMan システム・セットアップを随時起動することができます。

```
# /usr/sbin/checklist
```

カスタム・セットアップで一覧表示されるユーティリティは、システムにインストールされているサブセットによって決まります。たとえば、オプションのセキュリティ・サブセットがインストールされていない場合、「監査の設定」アプリケーションは表示されません。カスタム・セットアップで表示されるアプリケーションの一部を次に示します。

ライセンス管理	ディスクの設定	ネットワーク・セットアップ・ウィザード
DNS (BIND) の設定	NIS の設定	NFS の設定
アカウント管理	メールの設定	LAT の設定
NTP の設定	プリンタの設定	セキュリティの設定
監査の設定	DOP 特権の設定	Insight Manager
Prestoserve I/O アクセラレータの設定	GUI の選択	

また、SysMan システム・セットアップ・アプリケーションの多くは、文字セル端末でも利用できます。

システム・セットアップ全般の詳細については、『インストレーション・ガイド』、『システム管理ガイド』、『ネットワーク管理ガイド：接続編』、『ネットワーク管理ガイド：サービス編』、『*Software License Management*』、および setup(8) リファレンス・ページを参照してください。

2.4 Logical Storage Manager

Logical Storage Manager (LSM) は、オプションで用意されている、ホスト・ベースの統合型ディスク・ストレージ管理アプリケーションです。LSM は RAID 技術を採用しています。そのため、ストレージ・デバイスを構成して、データ損失からの保護、ディスク使用量の最大化、性能の改善、データ可用性の向上、およびストレージの管理を行うことができます。しかも、ディスク上のデータにアクセスするユーザやアプリケーションの妨げになりません。

LSM を使用すると、すべてのストレージ・デバイス (ディスク、パーティション、RAID セットなど) を、LSM ボリュームを作成する元となる柔軟性

に富んだストレージ・プールとして管理できます。ファイル・システム、データベース、およびアプリケーションを構成して、ディスク・パーティションの代わりに LSM ボリュームを使用します。パーティションの代わりに LSM ボリュームを使用する利点は、次のとおりです。

- データ損失からの保護

LSM は、データまたはデータとパリティ情報の組み合わせの複数のコピー (ミラー) を自動的に保存し、保守します。ストレージ・デバイスが故障した場合、LSM は次の方法でデータを保護します。

- ユーザやアプリケーションの妨げになったり、システムを停止したり、またはデータのバックアップおよびリストアを行うことなく、残っているミラーまたは残っているデータとパリティ情報を使用して動作を続行します。
- 障害が発生したストレージ・デバイスから、指定のスペア・ディスクまたは空きディスク・スペースへデータを転送し、再配置についてのメールをユーザに送信します。

また、LSM を使用すると、ブート・ディスク・パーティションを LSM ボリュームにカプセル化して、それらのボリュームにミラーを追加することもできます。その場合、本来のブート・ディスクの障害時にシステムをブートするためのブート・ディスク・パーティションのコピーが作成されます。

- ディスク使用量の最大化

LSM を構成してストレージ・デバイスをシームレスに結合させ、単一のストレージ要素としてユーザやアプリケーションに見せることができます。

- 性能の改善

LSM を構成して、データを同じサイズのユニットに分割し、それらのデータ・ユニットを 2 個以上のストレージ・デバイスに書き込むことができます。それらのストレージ・デバイスが異なる SCSI バスにある場合、LSM はデータ・ユニットを同時に書き込みます。

- データの可用性

TruCluster 環境で LSM を構成できます。TruCluster ソフトウェアは、複数の AlphaServer システムがネットワーク上の単一のシステムとして認識されるようにします。TruCluster ソフトウェアが実行されている

AlphaServer システムはクラスタのメンバとなり、リソースとデータ・ストレージを共有します。この共有によって、LSM などのアプリケーションは、そのアプリケーションが実行されているクラスタ・メンバが故障した場合でも、処理をそのまま続行できます。

LSM は、Tru64 UNIX CD-ROM に含まれているオプションのサブセットです。LSM は、Tru64 UNIX オペレーティング・システム・ソフトウェアのインストール時に一緒にインストールすることも、あるいは後でインストールすることもできます。

LSM のライセンスが無くても、ディスクとパーティションを結合させ、その記憶域を使用する LSM ボリュームを作成できます。LSM のそれ以外の機能を使用するためには、LSM のライセンスが必要です。HP のライセンス条件およびライセンスの購入については、最寄りの HP の営業所または正規代理店にお問い合わせください。

LSM の GUI である `lsmsa` は、Java 実行時環境を使用して、LSM コマンドを起動する方法を提供し、LSM オブジェクトの状態を監視できます。メイン・ウィンドウが表示されると、LSM オブジェクトが階層表示されます。オブジェクトをクリックすると、そのタイプのオブジェクトと、それらのオブジェクトの情報テーブルが表示されます。

2.5 システム管理ユーティリティ

SysMan アプリケーションは、インストール、構成、日常管理、監視、カーネルとプロセスの調整、およびストレージ管理のような管理タスクに関するアプリケーションを提供し、システムまたはネットワーク管理者の業務を容易にします。これらのアプリケーションは、`root` ユーザとしてログインした場合は、CDE フロント・パネルから SysMan ポップアップ・メニューを使用してアクセスできます。また、これらのアプリケーションの大半は、コマンド行でも同等の機能をサポートしています。

SysMan ユーティリティは、Common Desktop Environment (CDE) を利用するように設計されていますが、多くのユーティリティは、CDE 以外のその他のウィンドウや表示マネージャでも機能します。たとえば、次のコマンドは、「NIC (Network Interface Card) の設定」ツールを起動します。

```
# sysman interface
```

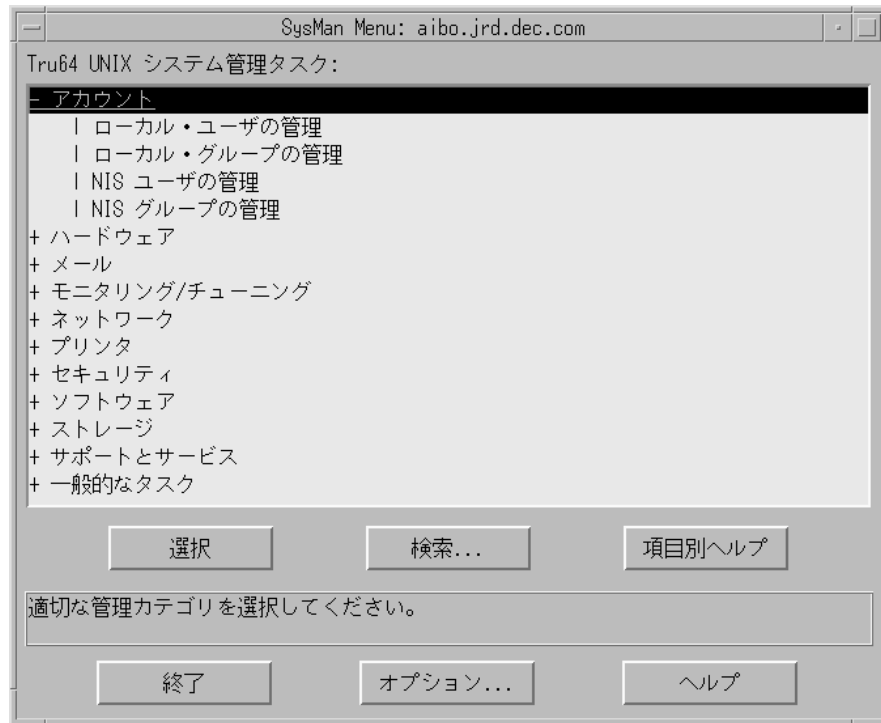
また、SysMan ユーティリティの多くは、文字セル端末でテキスト・ベースのインタフェースを使用して利用することもできます。

2.5.1 SysMan Menu

SysMan Menu (図 2-4) は、システム管理タスクのメニューをツリー状の階層で示します。管理タスクのカテゴリがブランチとして表示され、そのサブブランチ (リーフ) に実際のタスクが示されます。用意されているカテゴリには、メール、モニタとチューニング、ネットワーク機能、プリント機能、セキュリティ、ハードウェア、ソフトウェア、ストレージ、サポートおよびサービス、および一般タスクがあります。ユーザは、カテゴリ・ブランチを自由に展開して、サブブランチを表示することができます。サブブランチ (リーフ) を選択するとタスクが起動され、タスクを実行するためのダイアログ・ボックスが表示されます。

SysMan Menu は、`root` としてログインした場合に CDE フロント・パネルから起動されます。あるいは、`/usr/sbin/sysman` コマンドをコマンド行から入力すると直接起動できます。SysMan Menu は、CDE、HTML、または ASCII のテキスト環境で実行できるので、X11 対応ディスプレイ、Microsoft Windows 製品、Linux、または Macintosh オペレーティング・システムが稼働しているパソコン、あるいは文字セル端末で使用できます。

図 2-4: SysMan Menu



SysMan Menu は、ローカル・システムにインストールおよび構成されているオプションに応じて、次のようなアプリケーションを提供します。

- アカウント [accounts]
 - | ローカル・ユーザの管理 [users]
 - | ローカル・グループの管理 [groups]
 - | NIS ユーザの管理 [nis_users]
 - | NIS グループの管理 [nis_groups]
- ハードウェア [hardware]
 - | ハードウェア階層の参照 [hw_hierarchy]
 - | クラスタの参照 [hw_cluhierarchy]
 - | デバイス情報の参照 [hw_devices]
 - | CPU (Central Processing Unit) 情報の参照 [hw_cpus]
 - | CPU の管理 [hw_manage_cpus]
 - | Online Addition/Replacement (OLAR) ポリシー情報 [hw_olar_policy_info]
- メール [mail]
 - | メールの設定 [mailsetup]
 - | メール・アカウントの管理 [mailusradm]
- モニタリング/チューニング [monitoring]
 - | イベントの参照 [event_viewer]
 - | Insight Manager の設定 [imconfig]
 - クラス・スケジューリング [class_sched]
 - | クラス・スケジューラの設定 [class_setup]
 - | クラス・スケジューラの起動/再起動 [class_start]
 - | クラス・スケジューラの停止 [class_stop]
 - | 仮想メモリ (VM) 統計の参照 [vmstat]
 - | 入出力 (I/O) 統計の参照 [iostat]
 - | UPTIME 統計の参照 [uptime]

- ネットワーク [network]
 - | ネットワーク・セットアップ・ウィザード [net_wizard]
 - 基本ネットワーク・サービス [networkbasic]
 - | ATM (Asynchronous Transfer Mode) の設定 [atm]
 - | ネットワーク・カードの設定 [interface]
 - | スタティック・ルートの設定 (/etc/routes) [route]
 - | ルーティング・サービスの設定 (gated, routed, IP ルータ) [routing]
 - | ホスト・ファイルの設定 (/etc/hosts) [host]
 - | 等価ホスト・ファイルの設定 (/etc/hosts.equiv) [hosteq]
 - | リモート who サービスの設定 (rwhod) [rwhod]
 - | ネットワーク・ファイルの設定 (/etc/networks) [networks]
 - 追加ネットワーク・サービス [networkadditional]
 - Domain Name Service (DNS(BIND)) [dns]
 - | DNS サーバとしてシステムを設定 [dns_server]
 - | DNS クライアントとしてシステムを設定 [dns_client]
 - | このシステムの DNS の設定解除 [dns_deconfigure]
 - シリアル・ライン・ネットワーク [serial_line]
 - Point-to-Point Protocol (PPP) [ppp]
 - | オプション・ファイルの作成 [ppp_options]
 - | pap-secrets ファイルの修正 [pap]
 - | chap-secrets ファイルの修正 [chap]
 - | UNIX-to-UNIX コピー (uucp) 接続の設定 [uucp]
 - NTP (Network Time Protocol) [ntp]
 - | NTP クライアントとしてシステムを設定 [ntp_config]
 - | NTP のステータス参照 [ntp_status]
 - | NTP デーモンの起動/再起動 [ntp_start]
 - | NTP デーモンの停止 [ntp_stop]
 - NFS (Network File System) [nfs]
 - | NFS 設定状態の参照 [nfs_config_status]
 - | NFS クライアントとしてシステムを設定 [nfs_client]
 - | NFS クライアントとしてのシステム設定を解除 [nfs_deconfig_client]
 - | NFS サーバとしてシステムを設定 [nfs_server]
 - | NFS サーバとしてのシステム設定を解除 [nfs_deconfig_server]
 - | NFS デーモン・ステータスの参照 [nfs_daemon_status]
 - | NFS デーモンの起動/再起動 [nfs_start]
 - | NFS デーモンの停止 [nfs_stop]
 - | NIS (Network Information Service) の設定 [nis]
 - | LAT (Local Area Transport) の設定 [lat]
 - | DHCP サーバ(joind) の設定 [joind]
 - | ネットワーク・デーモンのステータスの参照 [dmnstatus]
 - | ネットワーク・サービスの起動/再起動 [inet_start]
 - | ネットワーク・サービスの停止 [inet_stop]
 - プリンタ [printers]
 - | ライン・プリンタの設定 [lprsetup]
 - セキュリティ [security]
 - | DOP (Division of Privilege) の設定 [dopconfig]
 - | DOP アクションの管理 [dopaction]
 - | セキュリティの設定 [secconfig]
 - | 監査の設定 [auditconfig]
 - ソフトウェア [software]
 - インストール [install]
 - | ソフトウェアのインストール [setldload]
 - | インストールされているソフトウェアの一覧 [setldlist]
 - | インストールされているソフトウェアの削除 [setldd]
 - | OS アップデート後のクリーンアップ (updadmin) [updadmin]
 - | ライセンス・データの登録 [lmfsetup]
- ストレージ [storage]
 - ファイルシステム管理ユーティリティ [filesystems]
 - 一般的なファイルシステム・ユーティリティ [generalfs]
 - | ファイルシステムのアンマウント [dismount]
 - | 現在マウントされているファイルシステムの表示 [df]
 - | ファイルシステムのマウント [mount]
 - | ローカル・ディレクトリ (/etc/exports) の共有 [export]
 - | ネットワーク・ディレクトリ (/etc/fstab) のマウント [net_mount]
 - Advanced File System (AdvFS) ユーティリティ [advfs]

```

| AdvFS ドメインの管理 [domain_manager]
| AdvFS ファイルの管理 [file_manager]
| AdvFS ドメインのデフラグメント [defrag]
| 新しい AdvFS ドメインの作成 [mkfdmn]
| 新しい AdvFS ファイル・セットの作成 [mkfset]
| AdvFS ドメインのファイルの修復 [salvage]
| AdvFS ドメインの修復 [verify]
- UNIX File System (UFS) ユーティリティ [ufs]
| 新しい UFS ファイルシステムの作成 [newfs]
- Logical Storage Manager (LSM) の管理 [lsm]
| Logical Storage Manager (LSM) の初期化 [volsetup]
| Logical Storage Manager (LSM) 管理 [lsmmgr]
| ブータブル・テープの作成 [boot_tape]
| SAN アプライアンス・ウィザードの識別 [idsanappl]
- サポートとサービス [support]
| エスカレーション・レポートの作成 [escalation]
| 構成レポートの作成 [config_report]
- 一般的なタスク [general_tasks]
| システムのシャットダウン [shutdown]
| クイック・セットアップ [quicksetup]
| Prestoserve ソフトウェアの設定 [presto]
| X ディスプレイ・マネージャの設定 [xsetup]
| クローニング・セットアップ情報 [cloneinfo]
| コマンド行インタフェース情報 [sysmancli]

```

[ufs] のようなメニュー項目ごとにキーボード・アクセラレータが用意されています。このアクセラレータによって、たとえば次のように実行することで、特定のツールをコマンド行から直接迅速に起動できます。

```
# sysman ufs
```

2.5.2 SysMan Station

SysMan Station (図 2-5) は、高度なプロファイル表示とシステムの物理的、論理的オブジェクトの状態表示を行います。このユーティリティは、Tru64 UNIX システムを管理するための基点となるように設計されています。

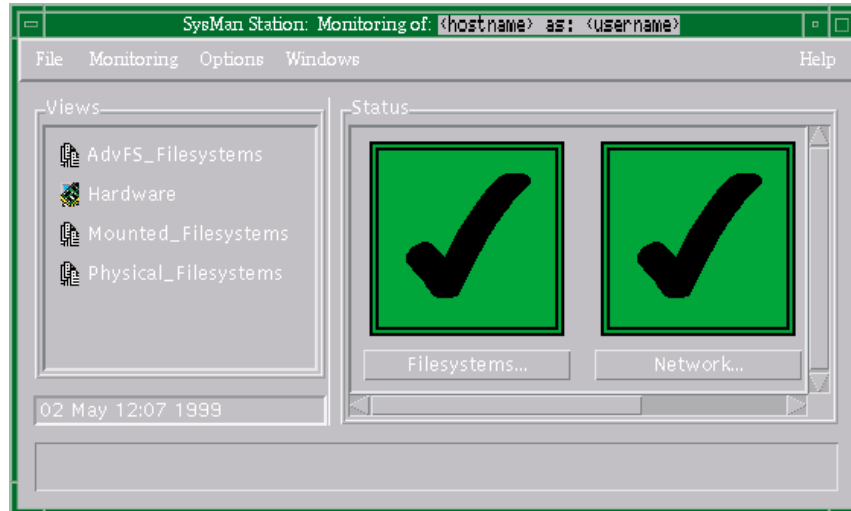
SysMan Station は、他の SysMan ツールを起動することによりタスクを実行します。アプリケーションは、SysMan Station の [Tools] メニューから直接起動することもできます。SysMan Station は、標準の Java 対応のディスプレイ上 (UNIX ワークステーションなど) または PC のブラウザ内で実行したり、PC 上にダウンロードして直接実行することができます。

SysMan Station を使用して、次のタスクを実行できます。

- システムまたはクラスタの状態監視
- システムまたはクラスタについての詳細情報の表示
- 管理作業の集中管理
- イベントの表示、および問題の原因となるイベントの追跡

root としてログインした場合，CDE フロント・パネルにあるアイコンから直接 SysMan Station を起動できます。あるいは，コマンド行から /usr/sbin/sms と入力して起動できます。

図 2-5: SysMan Station



2.5.3 CDE アプリケーション・マネージャ

Tru64 UNIX 上で CDE デスクトップを実行している場合は，CDE アプリケーション・マネージャにアクセスすることができます。CDE アプリケーション・マネージャの「システム管理」グループから，次の SysMan アプリケーションを起動できます。

- SysMan システム・セットアップ・チェック
- SysMan Station
- SysMan Menu
- 『SysMan の紹介』

このオンライン・ヘルプは，SysMan オンライン・ヘルプの概要を説明しています。

CDE アプリケーション・マネージャの「システム管理」フォルダにあるアイコンは，通常は SysMan Menu ユーティリティを起動します。たとえば，「システム設定」フォルダにある「DNS (BIND)」アイコンをクリックすると，SysMan Menu が起動し，SysMan Menu DNS オプションだけを表示し

ます。しかし、一部の機能に関しては、アイコンをクリックすると、X11 に準拠した GUI が起動されます。

これらの X11 準拠アプリケーションは、SysMan Menu から利用できる類似のツールとは明確に区別されます。これらの GUI は、オブジェクトのアイコン表示をドラッグ・アンド・ドロップして操作を実行する機能など、X Windows System ユーザ環境でのみ有効な機能を利用します。たとえば、CDE の「日常管理」フォルダ「アカウント・マネージャ」アイコンをクリックしても、SysMan Menu は起動されません。代わりに、X11 準拠の `dxaccounts` GUI が表示されます。

つまり、ユーザ環境あるいは好みに応じて、同一のタスクを実行するために使用可能な 2 個以上のインタフェースが存在する場合がありますということです。

「システム管理」グループには、次のカテゴリのシステム管理グループも含まれています。

- 「システム設定」グループには、Tru64 UNIX システムのインストール後に、システムを構成、およびセットアップするためのアプリケーションが含まれています。システムを一度セットアップしてしまえば、管理者がこれらのアプリケーションを定期的には使用することはありません。「システム設定」グループからは、次のアプリケーションを起動することができます。
 - BIND/DNS
 - CDE セットアップ
 - DHCP サーバ
 - ディスク構成
 - メール構成
 - NFS 構成
 - NIS セットアップ
 - PPP 構成
 - プリント構成
 - SLIP
 - LAT セットアップ

- 「日常管理」グループには、日常ベースで典型的な管理タスクを実行するためのアプリケーションが含まれています。このグループから次のアプリケーションを実行できます。
 - アカウント・マネージャ
 - アーカイバ
 - 監査マネージャ
 - ディスプレイ・ウィンドウ
 - イベント・ビューア
 - ファイル共有
 - ホスト・マネージャ
 - ライセンス・マネージャ
 - メール・ユーザ管理
 - パワー・マネージャ
 - シャットダウン
 - システム情報
- 「モニタリング/チューニング」グループには、システムをチューニングおよび監視するためのアプリケーションを提供します。このグループからは、次のアプリケーションを起動することができます。
 - カーネル・チューナ
 - プロセス・チューナ
 - システム情報
 - Insight Manager
- 「ソフトウェア管理」には、システム上で追加ソフトウェアを管理およびインストールするためのアプリケーションが含まれています。このグループからは、次のアプリケーションを起動することができます。
 - アップデート・インストレーション・クリーンアップ
 - ソフトウェア・インストール

- 「ストレージ管理」には、管理者によるファイル・システムの管理および監視を支援するためのアプリケーションが含まれています。このグループからは、次のアプリケーションを起動することができます。
 - Advanced File System
 - ブート・テープ
 - Logical Storage Manager
 - Prestoserve I/O Accelerator

2.6 性能とイベントの管理

Tru64 UNIX は、システムとそのイベントを監視し、管理するためのさまざまなユーティリティを用意しています。次のセクションでは、これらのユーティリティの概要を手短に説明します。これらのユーティリティの使用方法に関しては、『システム管理ガイド』と『インストール・ガイド』を参照してください。

2.6.1 HP Insight Manager

HP Insight Manager は、操作環境から独立して機能する Web ベースのユーティリティで、異機種混合環境でネットワークに接続されているデバイスの情報を入手することができます。システムの構成やそれらのコンポーネントおよび周辺機器についての情報を得ることができるだけでなく、場合によっては、資産管理、資産保全、ワークロード管理、イベント管理などの管理タスクを行うこともできます。

HP Insight Manager は、Web-Based Enterprise Services (WEBES) と呼ばれるシステム環境を管理するための手法を利用して、サポートするシステムを Web ブラウザから管理できるように SysMan とその他の Tru64 UNIX ユーティリティを統合するラッパーを提供します。

Tru64 UNIX システムでは、HP Insight Manager を使用してデバイスの詳細を調べることができますが、管理タスクを実行するには、SysMan Menu または SysMan Station を起動しなければなりません。Microsoft Windows が稼働している PC では、デバイスの詳細を調べると同時に、管理タスクを起動することもできます。

HP Insight Manager および HP Insight Management Agents についての詳細は、『システム管理ガイド』を参照してください。

2.6.2 Compaq Analyze

Compaq Analyze は、エラー・ログ・ファイルを絶えず調べて読み込むことによって、システムの自動バックグラウンド分析を実現しています。イベントを検出すると、分析規則に従って分析エンジンがエラー情報を収集し、システム管理者が決定した電子メール・アカウントにそれらの情報と分析結果を送信します。

Compaq Analyze がインストールされ、構成されると、Compaq Analyze はシステム・スタートアップ・プロシージャ内で自動的に起動され、バックグラウンド・プロセスを実行します。システム管理者は、必要に応じてこのプロセスの開始および停止を行なうことができます。Compaq Analyze がすでに実行中の場合、新しいプロセスは開始されませんが、単独で実行中のプロセスが、複数のユーザに対して、複数の GUI を表示することはできます。

Compaq Analyze は、HP の Web-Based Enterprise Services (WEBES) と連携して動作します。この機能により、障害のある CPU を Tru64 UNIX の OLAR 機能で通知するのが可能になります。

2.6.3 Monitoring Performance History ユーティリティ

Monitoring Performance History (MPH) ユーティリティは、Tru64 UNIX オペレーティング・システムとそのハードウェア環境の信頼性および可用性に関する正確で適切な情報を収集します。MPH は、エラー・ログとクラッシュ・ダンプの情報を 1 週間に 2 回コピーするシェル・スクリプトです。これらの情報は、分析のために電子メールを使用して HP に自動的に送られます。分析後、レポートが生成され、この情報のユーザ、たとえばソフトウェア開発部門、ハードウェア開発部門、製造部門、および HP のサービス部門に配布されます。

MPH は、バックグラウンド・タスクとして実行され、CPU のリソースをほとんど使用しません。MPH はユーザからは見えず、使用に際してトレーニングは不要です。また、システムの性能には影響しません。詳しい説明は、『システム管理ガイド』を参照してください。

2.6.4 sys_check ユーティリティ

sys_check は、システムの値とパラメータを記録するためのシステム性能パラメータの広範なダンプを作成し、システム・データの基準データを提供します。大きな変更をしたり、障害復旧作業を実行する前には、この

`sys_check` が役に立つ場合があります。 `sys_check` を実行すると、このユーティリティは標準出力へ HTML 形式の文書を出力します。 詳しい説明は、『システム管理ガイド』を参照してください。

2.6.5 X ベースのユーティリティ

システムの性能を複数の側面から素早くチェックできるように、いくつかの GUI ユーティリティが用意されています。 これらは X11 に準拠したインタフェースで表示される X ベースのユーティリティです。 CDE では、これらのユーティリティは CDE フロント・パネルにある Tool Drawer アイコンの下にまとめられています。 この Tool Drawer アイコンは、「アプリケーション・マネージャ」フォルダを表示します。 このフォルダには、次のサブフォルダに監視ユーティリティがあります。

- 「デスクトップ・ツール」

このフォルダには、CPU 使用量を監視する「システム負荷」やディスクごとのファイル・システムの領域の現在の状態を取得する「ディスクの使用状況」のような、単純なインタフェースが入っています。

- 「システム管理」

このフォルダには、監視のためのユーティリティが含まれている、次の2つのサブフォルダがあります。

- 「モニタリング/チューニング」

システムの設定のチェックと変更に役立つプロセス・チューナ `dxproctuner` やカーネル・チューナ `dxkerneltuner` などの GUI が含まれています。

- 「ツール」

常に出力を監視し、更新と表示に関するプリファレンスを設定する、`iostat` や `nestat` のようなコマンド行ユーティリティの GUI が含まれています。

他の GUI アプリケーションと同様に、システム情報に素早くアクセスしたり、システムの性能を監視できるように常に表示させておくため、これらのアイコンをシステム管理デスクトップに置いておくこともできます。 これらの GUI に関するプログラムは、`/usr/bin/X11` にあります。

詳しい説明は、『システム管理ガイド』を参照してください。

2.6.6 環境モニタリング

どのようなシステムでも、通気の不良、過熱状態、またはファンの故障のために、熱レベルが上昇する可能性があります。チェックを怠ると、予期しないシャットダウンにより、データが失われたり、システム自体が損傷を受けたりします。環境モニタリングは、ユーザに警告を発行して、順序正しくシステムをシャットダウンできるように、AlphaServer システムの熱の状態を監視します。

監視の主要部分は、ロード可能なカーネル・モジュールとそれに関連する API、Server System MIB サブエージェント・デーモン、envmond デーモン、envconfig ユーティリティの 4 つのコンポーネントで構成されています。詳しい説明は、『システム管理ガイド』を参照してください。

2.6.7 イベント・マネージャ

イベント・マネージャは、システムの構成要素がイベントおよび状態の情報を報告するために使用する複数のチャンネルに焦点を当てて監視します。これらのチャンネルには、システム・ロガー `syslog` とバイナリ・エラー・ロガー `binlog` が生成したものなどの、さまざまなログ・ファイルが含まれています。これらのチャンネルはそれぞれ、システムの一部のセグメントを監視します。たとえば、ディスクが故障した場合に発行されるプロセッサによるハードウェア・エラーの報告の監視や、一定のルーチン・タスクが正常に終了したかどうかの監視などです。

イベント・マネージャは、これらのイベントを単一のイベント・ストリームに組み合わせます。このイベント・ストリームは、システム管理者がリアルタイムで監視したり、記憶域から検索してイベントの履歴として表示させたりします。

イベント・マネージャの表示機能には、SysMan アプリケーションに統合された GUI イベント・ビューアと、管理者がさまざまな方法でイベントのチェック、ソート、編集ができるようにコマンド行ユーティリティのフル・セットが用意されています。

また、システム管理者は、選択された条件の自動通知を行うようにイベント・マネージャを設定することもできます。`syslog` や `binlog` のようなよく知られたイベント・チャンネルに置き換えるのではなく、イベント・マネージャはそれらを要約するので、それらのチャンネルは元の位置に残り、同一の

イベント・セットを処理しつづけます。同時に、イベント・マネージャはそれらのイベントへのアクセスをより簡単にします。

イベント・マネージャには、次の主要な機能があります。

- イベント情報の集約化
- ユーザとアプリケーションがイベントをポストし監視するための機能
- カスタム・イベント・チャンネルの要約化のサポート
- バイナリ・エラー・ログ・ファイルを変換するための DECevent との統合
- オンラインでの説明を含む、イベントの要約表示と詳細表示の選択
- イベントをシェル・スクリプトとコマンド行からポストし処理するための、コマンド行ユーティリティのフル・セット
- どのイベントを記録するかを制御し、同一のイベントによって使用される記憶域を最適化する、構成可能なイベント・ロガー
- 日常的にタスクの保管とパージを実行するための自動ログ・ファイル管理
- イベントをポストしアクセスするための構成可能な認証機能

詳しい説明は、『システム管理ガイド』を参照してください。

2.6.8 DECevent Translation and Reporting ユーティリティ

DECevent Translation and Reporting ユーティリティは、システム・イベント・ログ・ファイルを、編集された ASCII レポートに変換するエラー・ログ編集ユーティリティです。このユーティリティは、コマンド行と GUI の両方をサポートしています。

DECevent には、次の 2 つの主要な機能があります。

- 変換
DECevent は、イベントを、システム・イベント・エントリから生成した ASCII レポートに変換します。この変換は、ビットからテキストへの変換です。
- 分析と通知
DECevent は、分析によって、故障したデバイスを切り離せるように、常にシステムを監視しています。このユーティリティを使用して、潜在的な問題を適切な担当者に通知するように設定できます。

イベントのタイプ、日付、時間、イベント・エントリ番号ごとに、DECevent レポート情報を取得できます。生成するレポートの形式は、全情報の開示から簡略化された情報メッセージまで選択できます。

詳しい説明は、『*DECevent Translation and Reporting Utility*』を参照してください。

2.7 クラッシュ・ダンプの管理

突然システムが停止した場合には、メモリ内のデータとカーネル・イメージのすべてまたは一部が保存されることがあります。このような事態は、システム・クラッシュまたはパニックと呼ばれます。格納されたデータとステータス情報は、クラッシュ・ダンプと呼ばれます。クラッシュ・ダンプ後、システムはシャットダウンされてコンソール・プロンプトを表示します。ユーザは原因を究明し、問題を解決した後で、システムをリブートしなければなりません。通常、クラッシュ・ダンプ・ファイルとそれに関連したログ・ファイルは、テクニカル・サポート担当者が障害を分析するために必要とされます。

クラッシュ・ダンプを管理するには、クラッシュ・ダンプ・ファイルの作成方法を理解していなければなりません。また、クラッシュ・ダンプとクラッシュ・ダンプ・ファイルのためのスペースを確保しておかなければなりません。確保するスペースの量は、システムの構成とシステムに実行させるクラッシュ・ダンプのタイプにより異なります。

Tru64 UNIX のクラッシュ・ダンプ機能は広範囲に渡り、次の項目を指定することができます。

- 保持されるクラッシュ・ダンプの数
- クラッシュ・ダンプを無効にするオプション
- ダンプ・プロセスによるスワップ・パーティション (仮想メモリ) の使用方法
- ダンプが書き込まれる場所

この場所は、除外メモリ領域、ローカル・ディスク (スワップ領域を使用)、またはリモート・ホストのいずれかです。

- ダンプを圧縮するか、圧縮しないか

デバッガなどの分析ツールでは、非圧縮のダンプ・ファイルでなくとも構いません。

- ダンプ・ファイルのサイズ (内容を定義するか、部分ダンプを指定する)
- 継続可能ダンプ

システムを停止させることなく、メモリのスナップショットをダンプ・ファイルにコピーできます。ダンプ構成を計画する際にクラッシュ・ダンプ・サイズを推定するのに便利な方法です。

- 応答がないシステムでの強制クラッシュ・ダンプ
- ダンプ・ファイルの記録および保管の方法および位置

クラッシュ・ダンプの管理については、『システム管理ガイド』を参照してください。

2.8 ハードウェア管理

多くの場合、Tru64 UNIX は、ハードウェア管理機能を自動的に実行します。しかし、オペレーティング・システムには、デバイスの情報を表示し、必要に応じてハードウェア管理タスクを実行するためのツールが用意されています。

2.8.1 hwmgr ユーティリティ

ハードウェア管理用の `hwmgr` ユーティリティを使用すると、ディスクやテープのドライブ、プロセッサ、およびバスなどのハードウェアの構成要素が管理しやすくなります。

`hwmgr` ユーティリティには、次のようなオプションが用意されています。

- `-view`

情報を表示します。このオプションのバリエーションとして、ハードウェア識別子、デバイス特殊ファイル名、モデル、およびすべてのデバイスのシステム上の位置を表示する `-view devices` があります。また他に、現在のハードウェア・コンポーネントを階層表示する `-view hierarchy` もあります。

- `-flash`

ディスクを識別できるよう LED を点滅させます。ディスクは、SCSI バス番号、SCSI ターゲット番号、論理ユニット番号、またはそのデバイス特殊ファイル名によって識別されます。

- `-show component`

ハードウェア管理で登録されているものも含め、ハードウェア・コンポーネントを表示します。このコマンドは、出力として、そのコンポーネントが現在登録されていること、対応するデバイス・ノードがあること、クラスタ・ワイドのユニークな名前を持っていること、対応する属性が保存されていること、あるいは、ハードウェア・コンポーネント・データベースと一貫性がないことなどを示す一連の 1 文字フラグを返します。

詳しい説明は、`hwmgr(8)` リファレンス・ページを参照してください。

2.8.2 Dynamic Device Recognition

Dynamic Device Recognition (DDR) は、SCSI 装置の動作パラメータと属性を SCSI CAM 入出力サブシステムに記述するためのプログラムのフレームワークです。hwmgr ユーティリティがサポートしていない SCSI 装置で DDR を使用することによって、新しい装置を追加し、既存の装置を変更することができます。これを実行するには、`/sbin/DDR_config` というユーティリティと `/etc/DDR.dbase` というテキスト・データベースを使用して、インストール後に前述の入出力サブシステムを変更します。DDR.dbase データベースの修正後にシステムをリブートする必要はありません。

2.9 動的にロード可能なサブシステム

Tru64 UNIX は、システムでカーネル・サブシステムを一括化、ロード、および管理するための機能を提供しています。

Tru64 UNIX システムにインストールし実行するためのロード可能デバイス・ドライバの作成方法およびパッケージ方法については、デバイス・ドライバのドキュメントを参照してください。また、システムにインストールし実行するためのロード可能カーネル・サブシステムの作成方法およびパッケージ方法については、『*Writing Kernel Modules*』を参照してください。同様に、カーネル属性の動的構成およびチューニングをサポートするフレーム・ワークの詳細については、『*プログラミング・ガイド*』を参照してください。

2.10 動的システム構成

システムのチューニングを単純化するため、システム構成ファイルを編集したり、変更を有効化するために対象のカーネルを再構築またはリブートせずに、Tru64 UNIX では一定のカーネルの属性を変更できます。属性テーブルを使用することによって、Tru64 UNIX カーネル・サブシステムでも他社が開発したサブシステムであっても、各々のカーネル・サブシステムごとに、実行時に変更可能なカーネル属性を定義できます。カーネル属性が実行時の再構成をサポートしている場合は、`/sbin/sysconfig` コマンドに `-r` オプションを付けて使用することによって、ブート時以前にこの定義を実行します。カーネル属性データベースの `/etc/sysconfigtab` でエントリを追加または修正することによって、ブート時にこの定義を実行できます。

詳しい説明は、『システム管理ガイド』と『システムの構成とチューニング』を参照してください。

また、`dxkerneltuner` コマンドを使用してカーネル属性を変更することもできます。詳しい内容は、`dxkerneltuner(8)` リファレンス・ページを参照してください。

2.11 データレス管理サービス

Tru64 UNIX は、データレス管理サービス (DMS) をサポートしています。これによって、システムの `/(root)`、`/usr`、および `/var` パーティションを、DMS サーバに常駐させ、DMS クライアントにネットワーク経由で NFS マウントすることができます。`/` と `/var` のパーティションは、DMS クライアントごとに固有のものです。が、`/usr` パーティションは共用されます。DMS クライアントはローカルでスワップおよびダンプを行います。NFS を使用して追加のファイル・システムをマウントすることもできます。

DMS はディスクの使用量を節約し、また、DMS サーバ上で DMS クライアントの管理およびバックアップを行うことができるため、システム管理タスクを簡素化できます。

DMS についての詳細は、『*Sharing Software on a Local Area Network*』を参照してください。

ネットワーク機能

この章では、さまざまなネットワーク機能を実現する Tru64 UNIX の構成要素について説明します。次の各トピックについて説明します。

- インターネット・プロトコル — TCP/IP (3.1 節)
- ネットワークのサポート (3.2 節)
- アプリケーション・プログラミング・インタフェース (3.3 節)
- ネットワーク管理ソフトウェア (3.4 節)
- ネーミング・サービス (3.5 節)
- タイム・サービス (3.6 節)

3.1 インターネット・プロトコル

TCP/IPは、それぞれが異なるサービスを提供する種々のプロトコルをサポートしています。これらのプロトコルを使用すると、ネットワーク・ハードウェアに依存せずにネットワーク通信を行うことができます。TCP/IPプロトコル (図 3-1) は、次の 3 つのグループから構成されています。

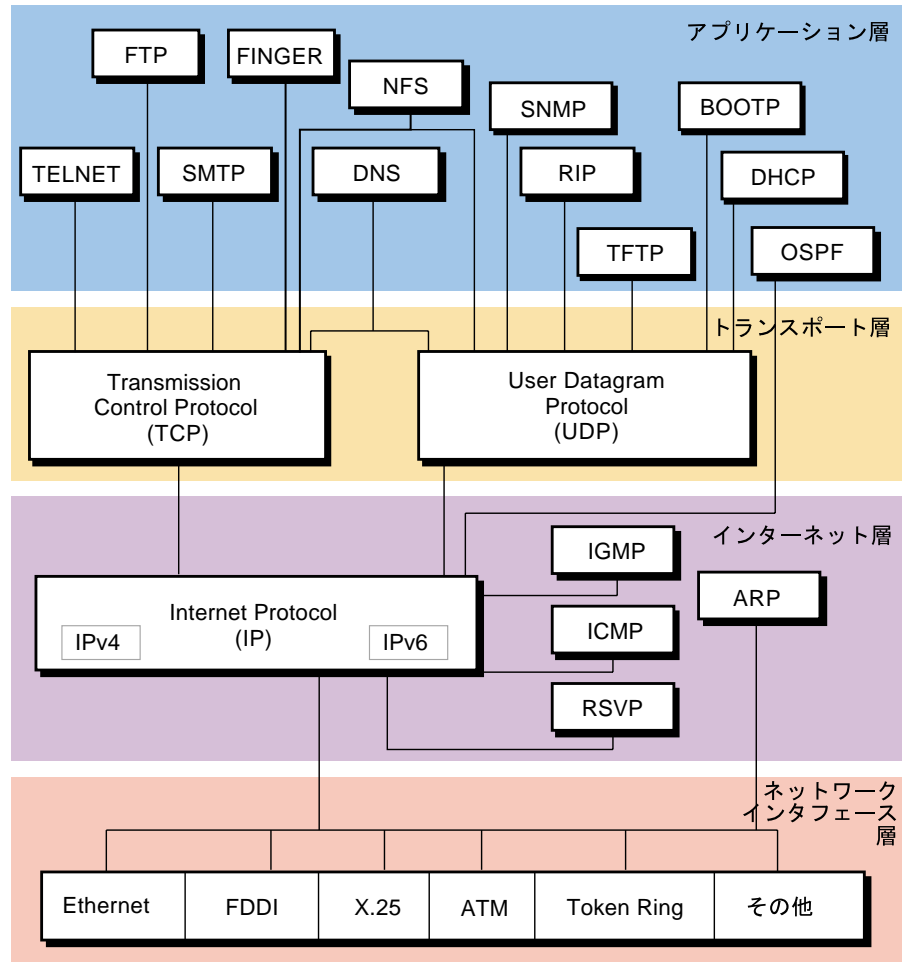
- アプリケーション・レベルのプロトコル
DNS, ルーティング・プロトコル (DVMRP, EGP, BGP, RIP, RIPng および OSPF), ファイル転送プロトコル (FTP), FINGER, TELNET, 簡易ファイル転送プロトコル (TFTP), シンプル・メール転送プロトコル (SMTP), シンプル・ネットワーク管理プロトコル (SNMP) など。
- トランスポート・レベルのプロトコル
ユーザ・データグラム・プロトコル (UDP), 伝送制御プロトコル (TCP) など。
- ネットワーク・レベルのプロトコル
アドレス解決プロトコル (ARP), インターネット制御メッセージ・プロトコル (ICMP), IPv6, Stateless Address Autoconfiguration, Neighbor Discovery, ICMPv6, Multicast Listener Discovery (MLD) など。

アプリケーション・プログラムは、ストリームまたはデータ・ブロックの形でメッセージを UDP および TCP の伝送プロトコルに送信します。これらのプロトコルはアプリケーションからデータを受信し、パケットに分割し、伝送ヘッダを追加し、そのパケットを次のプロトコル層であるインターネット層に伝送します。

インターネット層は、パケットを IP データグラム内に入れ、データグラム・ヘッダを付与し、データグラムを送信先ホストに直接送るかあるいは他のゲートウェイに送るか決定した後、ネットワーク・インタフェース層にこのデータグラムを渡します。ネットワーク・インタフェース層は、IP データグラムを受け取り、ネットワーク上の特定のハードウェアにフレームとして伝送します。

ネットワークが受信したフレームは、プロトコル層を逆方向に通過します。データがアプリケーション・レベルに戻る間に、個々の層で対応する各ヘッダ情報が取り除かれます。フレームは、ネットワーク・インタフェース層（たとえば Ethernet アダプタ）によって受信されます。ネットワーク・インタフェース層では物理層ヘッダが取り除かれ、データグラムがインターネット層に送信されます。インターネット層では、インターネット・プロトコルが IP ヘッダを取り除き、パケットをトランスポート層に送信します。トランスポート層では、TCP ヘッダまたは UDP ヘッダが取り除かれ、アプリケーション層へデータを送信します。

図 3-1: TCP/IP プロトコル



ZK-0819U-AIJ

3.1.1 アプリケーション・レベルのプロトコル

アプリケーションが別のホスト上のアプリケーションにデータを送信する必要がある場合、それらのアプリケーションは、トランスポート層のプロトコルに情報を送信して、情報伝達のための準備を行います。このようなプロトコルには、DNS、EGP、BGP、RIP、OSPF、FTP、NFS、TELNET、TFTP、FINGER、SMTP、SNMP などがあります。

3.1.1.1 DNS プロトコル

DNS (Domain Name Service) を使用すると、ドメイン内の 1 つまたは複数のホストを、ドメイン内の各ホストに対するネーム・サーバとして動作させることができます。DNS は UDP または TCP プロトコルの上で動作し、他のドメインから独立してローカル・ネットワーク内でホスト名を割り当てることができます。DNS で使用するプロトコルとしては UDP の方が適していますが、UDP からの応答が切り捨てられる場合は TCP を使用できます。

DNS についての詳細は、3.5.1 項を参照してください。

3.1.1.2 ルーティング・プロトコル

ルーティング・プロトコルによって、内部および外部の LAN 上のシステム間でルーティング情報を共用することができます。幾分時代後れになった EGP (外部ゲートウェイ・プロトコル) に加えて、Tru64 UNIX では、BGP (ボーダ・ゲートウェイ・プロトコル) と、RIP (ルーティング情報プロトコル) および OSPF (Open Shortest Path First Protocol) の両方を、NextHop Technologies 社の gated V4.0.6 ルーティング・デーモンの一部として提供します。gated についての詳細は 3.4.6 項を参照してください。

外部ゲートウェイ・プロトコル (EGP)

EGP (Exterior Gateway Protocol) は、独立したシステムの外部ゲートウェイが、他の独立したシステム上の外部ゲートウェイとルーティング情報を共有できるようにします。

ここでいう独立したシステムとは、1 人の管理者が責任を負うネットワークおよびゲートウェイからなる 1 つのグループです。各ゲートウェイが同一の独立システム内にある場合、それらを内部ゲートウェイと呼び、各ゲートウェイが複数の独立システムにまたがって存在する場合、それらを外部ゲートウェイと呼びます。EGP を使用してルーティング情報を交換するゲートウェイの集合を、EGP ピア (近隣ゲートウェイ) と呼びます。独立システムの各ゲートウェイは、EGP を使用して EGP ピアにルーティング情報を提供します。

EGP は、次のような方法で外部ゲートウェイがシステム間のリモート通信を提供できるようにします。

- 相手の外部ゲートウェイにルーティング情報の交換を要求する。

- EGP ピアの各ゲートウェイが応答しているかどうかを、絶えず確認する。
- ルーティング更新メッセージを EGP ピアに渡し、ルーティング情報を交換できるようにする。

EGP は、そのゲートウェイの独立システム内にある、完全にルーティング可能な送信先ネットワークだけを外部ゲートウェイに公示するよう、外部ゲートウェイを制限します。このため、EGP を使用する外部ゲートウェイは EGP ピアに情報を渡しますが、その EGP ピアに関するルーティング情報の公示は行いません。

EGP は、他のプロトコルからのルーティング更新メッセージに含まれる距離メトリックは解釈しません。EGP は、距離フィールドを使用して、パスが存在するかどうかを指定します。値が 255 の場合は、ネットワークが到達不可能であることを示します。2 つのルートが共に単一の独立システム内に含まれていない場合、この値は、2 つのルートのうち短い方を計算するために使用することはできません。このため、EGP をルーティング・アルゴリズムとして使用することはできません。したがって、外部ゲートウェイから任意のネットワークへの経路は 1 つだけになります。

EGP ルートは、`/etc/gated.conf` ファイル内で前もって定義されています。これは、ルーティング情報プロトコル (RIP) の場合とは対照的です。RIP は、動的にルートを再構成するインターネット・ネットワークの独立システム内で使用できます。EGP は、IP を基準プロトコルとして想定しています。詳細については、`gated(8)` を参照してください。

ボーダ・ゲートウェイ・プロトコル (BGP)

BGP (Border Gateway Protocol) は、複数のトランジット・オートノマス・システム間、あるいはトランジット・オートノマス・システムとスタブ・オートノマス・システム間でルーティング情報を交換するための外部ルーティング・プロトコルです。BGP は、EGP よりも多機能、柔軟、かつ少ない帯域幅で動作します。たとえば、BGP は、各ルートについての情報をより多く提供するためにパス属性を使用し、自由なトポロジによるルーティング・ループを防ぐために、ルートをトラバースした各オートノマス・システムの AS 番号など、十分な情報を提供するための AS (Autonomous System) パスをメンテナンスします。

EGP と同じように、BGP は内部セッションおよび外部セッションの両方をサポートします。外部ピアから受け取ったルートがパスの起点でそのピアの AS 番号を持つことを保証するために、外部ピアにルートを送ると、BGP はローカル AS 番号を AS パスの前に付加します。

内部の近隣ピアから受け取ったルートは、通常、AS パスに付加されたローカル AS 番号を持たず、起点となる内部近隣ピアが外部ピアからルートを受け取ったときにそのルートが持つと同じ AS パスを持ちます。パスに AS 番号を持たないルートを内部の近隣ピアから受け取ることがあります。これは、受け取ったルートが自身の AS の内部ピアであることを示しています。

Tru64 UNIX による BGP の実装では、バージョン 2、3、および 4 という 3 種類のバージョンの BGP プロトコルをサポートしています。BGP バージョン 2 および 3 は、機能の面で類似しています。これらのバージョンは、クラス化されたネットワーク経路だけを送信し、その AS パスは AS 番号の単純な配列です。BGP バージョン 4 は、十分に一般的なアドレスおよびマスクの経路を送信し、その AS パスは、異なる経路を統合した結果を表す、一定の構造を持っています。

ルーティング情報プロトコル RIP

RIP (Routing Information Protocol) は、ディスタンス・ベクトルあるいはローカル・ネットワークのための Bellman-Ford ルーティング・プロトコルのインプリメンテーションで、Tru64 UNIX では、NextHop Technologies 社の gated デーモンで提供しています。RIP は、ルータをアクティブ・ルータとパッシブ・ルータに分類します。アクティブ・ルータは他のルータにそれらのルートを公表しますが、パッシブ・ルータはルートの公表は行わず、受け取った公表データを基にそれらのルートを更新します。典型的には、ルータはアクティブ・モードで RIP を実行し、ホストはパッシブ・モードを使用します。

アクティブ・モードで RIP を実行しているルータは、アップデート情報を規定の間隔でブロードキャストします。各アップデート情報には、IP ネットワーク・アドレスとネットワークとの距離を表す値で構成された情報が含まれています。RIP は、デスティネーションとの距離をホップ数で測ります。ソースからデスティネーションの間のホップ数は、パスに沿ってデータグラムが通過するゲートウェイの数を表します。

たとえば、ルータに直接接続されているネットワークまでのホップ数は1です。別のゲートウェイを介して到達できるネットワークまでのホップ数は2です。2つのゲートウェイを介して到達できるネットワークまでのホップ数は3です。RIPは、最もホップ数が少ないパスを選択します。

もちろん、ネットワーク間のパスをホップ数で計算すると、常に最適な結果になるとは限りません。たとえば、3つのEthernet接続部分を通過するホップ数3のパスの方が、2つの低速度シリアル回線を通過するホップ数2のパスよりも速い場合があります。ネットワークとシリアル回線の転送速度の違いを補正するために、管理者は、低速のリンクに対してホップ数の多いパスを公表するようにRIPルータを構成することができます。

RIPng (Routing Information Protocol for IPv6)

RIPng (Routing Information Protocol for IPv6)は、ルーティング情報プロトコル(RIP)をベースとしています。RIPngにより、IPv6ベースのネットワークにおける経路の計算のための情報を、ルータ間で交換することができます。Tru64 UNIXでは、RIPngは`ip6rtrd`デーモンで提供しています。`ip6rtrd`デーモンは、アクティブ・モードのみサポートしており、IPv6ルータの機能を果たしているTru64 UNIXノード上でのみ実行されます。Tru64 UNIX IPv6ホストは、`nd6hostd`デーモンを実行します。

OSPF (Open Shortest Path First)

OSPF (Open Shortest Path First) ルーティング・プロトコルは、最短の第1パスか、または単一の自律型システムにあるルータ間にルーティング情報を配布するリンク状態内部ゲートウェイ・プロトコルです。多数のルータがある複雑なネットワークに適するように、OSPFは、単一の宛先に送られるパケットが2個以上のネットワーク・インタフェースによって同時に送信できるようにする等価マルチパス・ルーティングを実現します。

リンク状態プロトコルは、すべてのルータのリンク情報に関する公開情報を集めたAS全体のトポロジを記述したデータベースを各ルータが保守するように指示します。各ルータは、そのルータで利用できるインタフェースおよび到達可能な近隣ルータなどのローカル状態情報をAS全体に配布します。最低2つのルータが接続されている各マルチアクセス・ネットワークは、指定ルータとバックアップ・ルータを持っています。指定ルータは、リンク状態公開情報をマルチアクセス・ネットワークに流すとともに、その他の特別な

責任を持っています。指定ルータの概念によって、マルチアクセス・ネットワークで必要となる隣接ルータの数を減らせます。

OSPF はネットワークをエリアにグループ化します。エリア間で受け渡しされるルーティング情報は抽象化され、結果としてルーティング・トラフィックの顕著な低減が望めます。OSPF は 4 つのタイプのルートを使用します。4 つのタイプとは、優先する順に示すと次のようになります。

- intra エリア
- inter エリア
- タイプ 1 外部
- タイプ 2 外部

エリア内のパスには、同一のエリア内部の宛先が付いており、エリア間のパスには、外部の AS への宛先が付いています。

OSPF にタイプ 1 としてインポートされるルートは、外部メトリックスが OSPF メトリックスと直接比較できる EGP からのルートであると考えられます。ルーティングが決定されると、OSPF は AS 境界ルータへの外部コストを外部メトリックに追加します。

タイプ 2 ASE は、メトリックスが OSPF メトリックスと比較できない EGP に対して使用されます。この場合、AS 境界ルータへの内部 OSPF コストのみがルーティングの決定に使用されます。

各ルータは、ルータ自身をルート (root) とする最短パス・ツリーをトポロジ・データベースから作成します。この最短パス・ツリーは、AS における各デスティネーションへのルートを規定します。このツリーの葉として表されるのは、外部から引き出されるルーティング情報です。リンク状態公開情報のフォーマットは、外部ソースから取得された情報と内部ルータから取得された情報を識別するため、ルートのソースあるいは信頼性にあいまいさは含んでいません。EGP あるいは BGP など、外部から取得したルーティング情報は、オートノマス・システム全体に透過的に渡され、OSPF 内部から得られたデータとは別に保管されます。オートノマス・システムの境界上のルータ間で追加情報の受け渡しが可能になるように、各外部ルートには、公開ルータによって標識を付けることができます。

3.1.1.3 ファイル転送プロトコル (FTP)

FTP (File Transfer Protocol) を使用すると、ホスト間でファイルを転送することができます。FTP は、リモート・ディレクトリの表示、現在のリモート・ディレクトリの変更、リモート・ディレクトリの作成および削除、複数ファイルの転送などの機能を提供します。FTP は、ユーザおよびアカウント・パスワードを外部ホストへ渡すことによって、転送の際のセキュリティを保証します。FTP により、ユーザ指向の対話的セッションが可能になります。

FTP は、信頼性の高いストリーム転送 (TCP/IP) を使用してファイルを転送し、TELNET 型接続を使用してコマンドおよびその応答を転送します。FTP は、ASCII、IMAGE、Local-8 などの、基本的なファイル・フォーマットを認識します。TCP/IP は、ftp ユーザ・コマンドと ftpd サーバ・コマンドで FTP を実現します。

3.1.1.4 UDP トランスポート上のネットワーク・ファイル・システム・プロトコル

ネットワーク・ファイル・システム (NFS) は、標準の UNIX システム・コールによるファイルへのアクセスを提供します。NFS を使用することにより、各プログラムがネットワーク経由でファイルにアクセスすることができます。NFS は UDP トランスポート層を使用するので、データグラムの変換に対する対処が必要になります。NFS は、適切な時間内に応答がない場合、要求を再転送することによりこの問題に対処します。

この場合、要求によっては問題なくサーバで再実行されますが、たとえばファイル削除要求でその要求に対する応答が損失した場合、要求を再実行するとエラーが発生します。この場合 2 回目の要求が実行されると、ファイルが存在しないためにサーバはエラーを返します。NFS サーバは、そのような応答を保持しておき、繰り返し要求がある場合はその応答を再転送します。

このプロトコルは、サーバが他の状態情報を必要としないように設計されています。また、サーバ・デーモンの複数のコピーを実行することによりサーバの性能を向上させることが可能です。さらに、クライアントあるいはサーバに特別なコードがなくても、サーバのクラッシュが許容されます。

NFS についての詳細は、4.6 節を参照してください。

3.1.1.5 TCP トランスポート上のネットワーク・ファイル・システム・プロトコル

Tru64 UNIX は、TCP 経由での NFS をサポートします。LAN においては UDPの方が依然好まれるかもしれませんが、広域エリア、過密エリア、ある

いは損失の見られるネットワークにおけるNFS に対しては、TCP の方がよりすぐれた性能が得られます。

UDP コード・パスに対して行われたいくつかの性能の最適化処理を維持するために、別々のスレッドが使用されます。nfsiod デーモンは、複数のプロセスへ枝分かれする代りにカーネル・スレッドを新しく作成するように変更されています。各 nfsiod スレッドは、UDP あるいは TCP マウントを処理できるので、nfsiod コマンドには 1 つの引数が指定できます。

NFS についての詳細は、4.6 節を参照してください。

3.1.1.6 TELNET プロトコル

TELNET プロトコルは、端末デバイスと端末指向プロセスが互いに接続をするための標準的な方法を提供します。TELNET は、ユーザがリモート・ホストにログインするための端末エミュレーション・プログラムで使用されます。また、端末間通信やプロセス間通信にも使用することができます。

TCP/IP は、telnet ユーザ・コマンドと telnetd サーバ・コマンドで TELNET を実現します。

3.1.1.7 簡易ファイル転送プロトコル (TFTP)

TFTP (Trivial File Transfer Protocol) を使用すると、外部ホストのファイルに対して読み取り/書き込みを行うことができます。FTP と同様に、TFTP はファイルを 8 ビットの NETASCII 文字または 8 ビットのバイナリ・データとして転送することができます。TFTP が FTP と異なる点は、外部ホストでディレクトリの表示および変更ができないこと、および、パスワード保護などのセキュリティのための機能がないことです。通常は、公用ディレクトリにおけるデータの書き込みおよび検索が可能です。

TCP/IP は、tftp ユーザ・コマンドと tftpd サーバ・コマンドで TFTP を実現します。

3.1.1.8 FINGER プロトコル

FINGER プロトコルは、finger コマンドと fingerd デーモン間のインタフェースを提供するアプリケーション・レベルのインターネット・プロトコルです。fingerd デーモンは、指定されたリモート・ホストに現在ログインしているユーザの情報を返します。特定のホストでユーザを指定して finger コマンドを実行すると、そのユーザについての情報を得ることができ

ます。FINGER プロトコルは、リモート・ホストと要求側ホストの両方に存在している必要があります。FINGER は、TCP プロトコルを使用します。

3.1.1.9 シンプル・メール転送プロトコル (SMTP)

SMTP (Simple Mail Transfer Protocol) は、インターネットに接続されているマシン間でメールを交換するための標準です。このプロトコルは、電子メールを交換するために 2 つのホスト間で送信する制御メッセージのフォーマットを指定します。

名前が示すように、SMTP のしくみと用途は単純です。SMTP は、信頼性の高い効率的なメール転送システムを提供します。SMTP はメール・インタフェースを指定しません。

3.1.1.10 シンプル・ネットワーク管理プロトコル (SNMP)

SNMP (Simple Network Management Protocol) は、ネットワーク管理情報を交換するための、インターネット標準プロトコルです。SNMP エージェントは、管理情報ベース (MIB) にアクセスすることによって、ローカルあるいはリモートのネットワーク・マネージャに情報を提供します。SNMP エージェントの構成、セキュリティ、クラスタ・サポート、およびサポートする RFC については、`snmpd(8)` を参照してください。

Tru64 UNIX は、業界標準 (IETF RFC) の MIB と HP 独自の MIB の両方をサポートします。詳細は 3.4.5 項を参照してください。HP 独自の MIB 仕様は、`/usr/share/sysman/mibs` ディレクトリに置かれています。

MIB サポートを提供するデーモンは、`/usr/sbin/os_mibs`、`/usr/sbin/srvSystem_mib`、`/usr/sbin/svrMgt_mib`、`/usr/sbin/cpq_mibs`、および `/usr/sbin/clu_mibs` です。

Tru64 UNIX には、SNMP MIB データに Web ベース・インタフェースでアクセスするための HP Insight Manager が含まれています。HP Insight Manager については 2.6.1 項を参照してください。

拡張可能 SNMP プログラミング・インタフェースについての詳細は、3.3.6 項を参照してください。

3.1.1.11 POP3

POP3 (Post Office Protocol) は Qualcomm 社のクライアント/サーバ・プロトコルで、ユーザはこれを使用して電子メールをメール・サーバからリモー

ト・クライアントにダウンロードすることができます。メッセージがサーバに配信された後、ユーザがサーバに接続して、メッセージをクライアント・マシン (Windows, MacOS, UNIX, あるいはその他のオペレーティング・システムを実行しているデスクトップまたはラップトップ・コンピュータ) にダウンロードします。それ以降、すべてのメッセージは、クライアント・マシンおよび環境でローカルに処理されます。これは、ISP (インターネット・サービス・プロバイダ) が今日、顧客に電子メール・サービスを提供するために広く使用しているプロトコルです。詳細については、『ネットワーク管理ガイド：サービス編』を参照してください。

Tru64 UNIX は、POP3 Version 3.0.2 をサポートしています。このプロトコルは、RFC 1939 で定義されています。

3.1.1.12 IMAP4

IMAP4 (Internet Message Access Protocol) は、クライアント/サーバ・プロトコルで、カーネギー・メロン大学の Cyrus IMAP4 Revision 1 サーバをベースとしており、メール・クライアントは、これによりサーバ上のメール・メッセージにアクセスすることができます。IMAP4 を使用して、ユーザはサーバにログインすることなく、自分のメール・フォルダにアクセスし、内容をリモートに操作できます。クライアントはこのプロトコルを使用して、メール・フォルダの作成、削除、名前の変更、新しいメッセージのチェックと古いメッセージの削除、および選択したメッセージのローカル表示が行えます。加えて、ユーザはメッセージを属性によって選択したり、RFC 822 および MIME 形式でメッセージを解析したりすることができます。詳細については、『ネットワーク管理ガイド：サービス編』を参照してください。

Tru64 UNIX は、IMAP4 Version 1.6.24 をサポートします。このプロトコルは、RFC 2060 で定義されています。

3.1.1.13 RSVP

RSVP (Resource ReSerVation Protocol) は、RFC 2205 で定義されているインターネット・ネットワーク層です。これは、ネットワーク帯域幅管理コンポーネントの 1 つで、特定のアプリケーション・データ・ストリーム、またはフロー (シンプレックス・ユニキャストまたはマルチキャスト) に対する quality-of-service 要求をネットワーク経由で送受信できるようなメカニズムを提供します。要求が受け入れられると、特定の量のネットワーク帯域幅がフロー用に確保されます。

ビデオや音声について、省略時のベスト・エフォート配信が受け付けられない場合、アプリケーションは RAPI (RSVP API) を使用して拡張 quality-of-service を要求します。アプリケーションが要求できる quality-of-service のタイプは、Internet Integrated Services (RFC 1633 および RFC 2210) で定義されています。

Tru64 UNIX における RSVP は、FDDI と Ethernet インタフェース、およびユニキャストとマルチキャストのデータ・フローをサポートします。詳細については、『ネットワーク・プログラミング・ガイド』を参照してください。

3.1.2 トランスポート・レベルのプロトコル

TCP/IP のトランスポート・レベルのプロトコル (UDP と TCP) は、アプリケーション・プログラムが他のアプリケーション・プログラムと通信できるようにします。ユーザ・データグラム・プロトコル (UDP) と伝送制御プロトコル (TCP) は、インターネット・ホスト間の接続を行うトランスポート・レベルの基本的なプロトコルです。アプリケーションがメッセージをトランスポート層に送信する場合には、UDP と TCP は、メッセージを複数のパケットに分割し、送信先アドレスを含むパケット・ヘッダを付加した後、それらのパケットをネットワーク層に送信します。

UDP を使用してデータグラム接続を行い、TCP を使用してストリーム接続を行うプロトコルやアプリケーションもあります。UDP と TCP は、ソケット・インタフェースにより実現されます。

3.1.2.1 ユーザ・データグラム・プロトコル (UDP)

UDP (User Datagram Protocol) は、インターネット・ホスト上のアプリケーションと通信するデータグラムを使用します。UDP は、正の整数で識別されるマシン内の抽象的な送信先を示す送信先プロトコル・ポートを使用して、ホスト上の複数の送信先の 1 つにメッセージを送信します。プロトコル・ポートは、受信側ホスト上のアプリケーションがメッセージを検索し終わるまで、受信したメッセージをキューに保持します。

UDP は、インターネット・プロトコルによってデータグラムを送信し、IP と同じコネクションレス型のメッセージ送信機能を提供します。UDP は、データグラム転送や二重化の保護については保証しませんが、メッセージのソース・ポート番号およびデスティネーション・ポート番号を送信者が指定できるようにし、また、データとヘッダ両方のチェックサムを計算しま

す。これら 2 つの機能によって、アプリケーション間におけるメッセージの正確な送受信が保証されます。

3.1.2.2 伝送制御プロトコル (TCP)

伝送制御プロトコル (TCP) は、インターネット・ホスト間で、信頼性の高いデータ・ストリーム引き渡しを提供します。UDP と同様に、TCP は基準プロトコルである IP を使用してデータグラムを送信し、処理ポート間でのデータグラムの連続的なブロック伝送をサポートします。TCP が UDP と異なる点は、信頼性の高いメッセージ転送を提供することです。TCP は、損傷、損失、および二重アクセスなしに、送信順に受信プロセスにデータを引き渡すことを保証します。このように TCP では伝送の信頼性が保証されているため、アプリケーション・プログラム内に通信保護機能を用意する必要はありません。

TCP および UDP は共に、他のホストのアプリケーションとの間でプログラムがメッセージを送受信できるようにします。また、メッセージの送信先を識別するために、ローカル・ホスト上のプロトコル・ポートを使用します。TCP 再送タイムアウト値は、往復時間に基づいて、個々の接続ごとに動的に決定されます。

TCP には次のような特徴があります。

- 基本的なデータ転送

TCP は、ユーザ間で互いに、8 ビット・オクテットの連続したストリームを転送するために、一定のバイト数ずつまとめてパケットにし、インターネット・システムを経由して転送します。通常、TCP がパケットのブロック化および転送に最適な時間を決定します。

- 信頼性

TCP は、インターネットによって損傷、損失、重複アクセス、あるいは送信順と異なる引き渡しなどが発生したデータを修復します。TCP は、転送する個々のオクテットにシーケンス番号を割り当て、受信側 TCP に対して肯定応答 (ACK) を要求することによって、この信頼性を達成しています。タイムアウト時間内に ACK が受信されない場合には、データは再送されます。受信側では、シーケンス番号を使用して、セグメントを送信順に並べ、重複をなくします。伝送される個々のセグメントには損傷を検出するためにチェックサムが追加され、受信側でそれをチェックし、損傷したセグメントは廃棄します。

- フロー制御

送信するデータ量を制御するために、TCP は次の情報を返します。

- 次に受け入れる予定のシーケンス番号

このシーケンス番号は、正常に受信された最新のセグメントの番号より、常に大きな値になります。

- 別途許可されるまでに送信側が伝送できるオクテット数

- 多重化

TCP 通信機能は、単一ホスト内の多数のプロセスが同時に使用することができます。TCP は、個々のホスト内のアドレス (ポート) に関する情報を保持しています。TCP は、ネットワーク・アドレスとホスト・アドレスにポート番号を結合し、各接続の終端 (ソケット) を一意に識別します。1 組のソケットによって、個々の接続が一意に識別できます。

- 接続

TCP は、各データ・ストリームの状態情報を初期化し、維持します。この状態情報とソケット、シーケンス番号、ウィンドウのサイズの組み合わせを接続と呼びます。各接続は一意に識別されます。

3.1.3 インターネット・ネットワーク・レベルのプロトコル

インターネットのネットワーク・レベルのプロトコル (IP, ARP, ICMP) は、コンピュータ間の通信を処理します。これらのプロトコルによって、転送要求の送受信が行われるとともに、ネットワーク・レベルでの制御が行われます。

3.1.3.1 IPv4

IPv4 (Internet Protocol Version 4) は、ネットワーク・レベルの主要なプロトコルであり、インターネット経由で送信されるデータの形式を定めたものです。また、IPv4 はパケット処理とエラー処理についても指定しています。

IP は、個々のパケットを独立して扱うコネクションレス型であり、引き渡しあるいはパケットの到着順序を保証しないため信頼性は低くなります。

IP は、ネットワーク・インタフェース・レベルのプロトコルにインタフェースを提供します。物理的に接続されたネットワークでは、情報はヘッダとデータという形態で転送されます。IP は、シーケンス情報や制御情報の

他に、ソース・ホストのアドレスを含むインターネット・データグラムを使用します。

発信パケットの先頭には IP ヘッダが自動的に付けられますが、より高レベルのプロトコルに送信される前に IP ヘッダは削除されます。IP は、インターネット・ネットワーク内にあるホストに対して、地球規模でアドレスを指定することができます。

ただし、IP は、送信側ホスト、受信側ホスト、中間ホストからの肯定応答を要求しないため、信頼性の高い通信機能ではありません。

IP パケットの全長は、個々のインタフェースごとに独立して構成することができます。あるインタフェースにとって大きすぎるパケットは、複数のより小さいパケット (断片) に分割できます。元のパケットは送信先で断片から再構成されます。

IPv4 マルチキャストイング

Tru64 UNIX は、RFC 1112 に記述されている、LAN における IPv4 マルチキャストイングをサポートします。また、Tru64 UNIX は、Version 3.5 の IPv4 マルチキャスト・カーネルおよび Version 3.6 の DVMRP (Distance Vector Multicast Routing Protocol) の `mrouted` インプリメンテーションをサポートしています。

IPv4 ブロードキャストイングとは違い、IPv4 マルチキャストイングの場合、パケットを受け取るように構成されているクライアントによってのみ、パケットをネットワークから取り出すことができます。パケットはハードウェア・レベルで受け入れたり拒否したりするため、処理のためのオーバーヘッドが減少します。IPv4 マルチキャストイングの場合、アプリケーションが同一データを複数の送信先へ送信するのに別々のパケットを使用する必要がないため、ネットワーク帯域幅をあまり消費しません。IPv4 マルチキャストイングでは、1 つのパケットですべてのホストへ送信を行います。

このため、IPv4 マルチキャストイングは、ビデオ会議アプリケーションや株式取引相場に関するアプリケーションのように、情報が更新されるたびに最新情報を送信するようなアプリケーションに対して有効です。

IPv4 マルチキャストイングのコードはパブリック・ドメインのものを使用し、すべての Ethernet および FDDI アダプタがサポートされています。

3.1.3.2 IPv6

IPv6 (Internet Protocol Version 6) は、新しいネットワーク・レベル・プロトコルで、インターネット上で送信されるデータの形式を定めたものです。また、IPv6 はパケット処理とエラー処理の仕様も定めています。

IPv6 は次のようなネットワーク・インタフェースでサポートされています。

- Ethernet
- FDDI (Fiber Distributed Data Interface)
- PPP (Point-to-Point Protocol)
- IP トンネル
- Loopback

Neighbor Discovery

同じリンク上にある IPv6 ノードが Neighbor Discovery を使用すると、お互いの存在を知り、相互のリンク層アドレスを特定し、ルータを探し、さらにアクティブな近隣ノードまでの経路に関する到達可能性情報を保持することができます。

ノード (ホストおよびルータ) は、Neighbor Discovery プロトコルを使用して、接続されているリンク上に存在することがわかっている近隣ノードのリンク層アドレスを特定したり、無効になったキャッシュの値を速やかに消去します。ホストは、パケットの転送を行う用意のある近隣ルータを探す際にも Neighbor Discovery プロトコルを使用します。また、ノードは、近隣ノードのうちどれが到達可能でどれが到達不可能かを積極的に把握したり、リンク層アドレスの変更を検出するために、このプロトコルを使用することもあります。

Multicast Listener Discovery

Multicast Listener Discovery (MLD) によって、IPv6 ルータは、自分が直接接続されているリンク上にあるマルチキャスト・リスナ (マルチキャスト・パケットの受信を希望するノード) の存在、およびそれらの近隣ノードが具体的にはどのようなマルチキャスト・アドレスを必要としているかを知ることができます。その情報が、ルータが使用しているマルチキャスト・ルーティング・プロトコルに提供され、その結果、受信を希望するノードの存在するすべてのリンクにマルチキャスト・パケットが確実に引き渡されます。

ICMPv6

ICMPv6 (Internet Control Message Protocol Version 6) は、IPv6 のすべての実装に必要な構成要素です。ICMPv6 は、IPv6 のエラー・メッセージおよび制御メッセージを処理します。

ICMPv6 の機能は次のとおりです。

- デスティネーションが動作中で、パケットが到達可能かどうか調べる。
- データグラム・ヘッダのパラメータに誤りがあれば通知する。
- Multicast Listener Discovery を実行する。
- IPv6 アドレスを取得する。
- ルータの通知を要求し、処理する。
- 重複アドレスの検出を行う。
- 近隣ノードのリンク層アドレスを要求し、処理する。
- トランスポート・レベルの到達可能性情報を提供する。
- ルーティング情報を更新する。

ICMPv6 は、通信環境の問題に関するフィードバックを行いますが、IPv6 の信頼性を高めるものではありません。つまり、ICMPv6 は、IPv6 パケットが確実に引き渡されることを保証するものではありません。また、IPv6 パケットが引き渡されなかった場合や正しく引き渡されなかった場合に、ICMPv6 メッセージを送信元のホストに戻すことを保証するものでもありません。

Mobile IPv6

IPv6 は、拡張ヘッダ構造、アドレス自動構成、セキュリティ (IPsec)、トンネリングなどの機能によって機動性を確保するように設計されています。Mobile IPv6 はこれらの機能の上に構築され、IP アドレスの変更を伴わないでモバイル・ノードをあるリンク・ポイントから別のリンク・ポイントへ移動するための操作を定義します。これにより、モバイル・ノードが別のネットワーク上にあっても、パケットをそのモバイル・ノードに対して送受信することができます。

Mobile IPv6 の実装は *Mobility Support in IPv6* をベースにしており、次の機能をサポートします。

- 新しい IPv6 デスティネーション・オプションおよびメッセージ

- コレスポンデント・ノード操作

Mobile IPv6 の計画，構成，監視については，『ネットワーク管理ガイド：接続編』を参照してください。

Mobility Support in IPv6 のドラフトは次の URL で参照できます。

<http://www.ietf.org/internet-drafts/draft-ietf-mobileip-ipv6-18.txt>

IPv6 をサポートするコマンドおよびデーモン

次のコマンドおよびデーモンは，IPv6 と IPV4 の両方の環境で動作するように修正されています。

/sbin/dump	/sbin/ifconfig	/sbin/init.d/inet	/sbin/init.d/route
/sbin/named	/sbin/ping	/sbin/restore	/sbin/route
/usr/bin/finger	/usr/bin/ftp	/usr/bin/nrdist	/usr/bin/rcp
/usr/bin/rdist	/usr/bin/rdistd	/usr/bin/telnet	/usr/sbin/dump
/usr/sbin/ftpd	/usr/sbin/ifconfig	/usr/sbin/inetd	/usr/sbin/named
/usr/sbin/nd6hostd	/usr/sbin/netstat	/usr/sbin/ping	/usr/sbin/pppd
/usr/sbin/rcinet	/usr/sbin/rdump	/usr/sbin/restore	/usr/sbin/rexecd
/usr/sbin/rlogind	/usr/sbin/route	/usr/sbin/rrestore	/usr/sbin/rshd
/usr/sbin/rsvpd	/usr/sbin/rwhod	/usr/sbin/tcpdump	/usr/sbin/telnetd
/usr/sbin/traceroute			

3.1.3.3 IPsec (Internet Protocol Security)

IPsec (Internet Protocol Security) は，IPv4 (Internet Protocol Version 4) および IPv6 (Internet Protocol Version 6) のための，相互運用可能で高品質の暗号に基づくセキュリティを提供するセキュリティ・フレームワークです。

次のようなセキュリティ・サービスを提供します。

- アクセス制御

認証されたネットワーク，システム，アプリケーションのみがホスト，保管データ，セキュリティ・ゲートウェイの後ろのネットワーク，あるいはネットワークの帯域幅などのアクセスすることができます。

- コネクションレス完全性

トラフィック上の IP パケットの順序に関係なく，個々の IP パケットの変更を自動的に検出します。

- データの素性の証明

所定のソースからのメッセージが確かにそのソースから届いているか確認します。

- 再送攻撃からの保護

パケット送信時に送信側でカウントアップされるシーケンス番号を受信側でも確認し、不正なパケットを排除する。

- 秘匿性

暗号化によって、アプリケーション・レベルのデータのセキュリティを保護します。

- 限定されたトラフィック・フローの秘匿性

発信元アドレス、着信先アドレス、メッセージ長、通信頻度などを暗号化して、転送データ(トンネル・モード)のセキュリティを保護します。

これらのサービスは Internet Protocol (IP) 層で提供され、IP およびより上位のプロトコルで保護機能を提供します。

IP をネットワーク層のプロトコルとして使用するほとんどのシステムでは、データグラムは常に他人の目に触れるオープンな形で伝達され受信されます。プライベートなネットワークであれば、このことは大きな問題にはなりません。しかし、パブリックなネットワークにおいては、重大なセキュリティ問題になる可能性があります。アプリケーションによっては、SOCKS、SSL (Secure Socket Layer)、Secure HTTP (S-HTTP)、Secure Mail (S-MIME) など、独自のセキュリティ機能を提供することによりこの問題を解決するものもあります。IPsec は、インターネットおよびイントラネットを流れるすべてのデータを保護することが可能です。

この実装では、次のような IPv4 および IPv6 に対する完全な IPsec プロトコルをサポートします。

- IPv4 および IPv6 対応の AH (Authentication Header) ヘッダ
- IPv4 および IPv6 対応の ESP (Encapsulating Security Payload) ヘッダ
- AH および ESP に対する HMAC-MD5 および HMAC-SHA-1 証明アルゴリズム
- DES (Data Encryption Standard)、3DES、AES (Advanced Encryption Standard) 暗号化アルゴリズム

- Deflate アルゴリズムによる IP 圧縮 (IPCOMP)
- IPv4 および IPv6 における AH および ESP に対するトンネルおよび
トランスポート・モード
- RFC 2401 で必要とされる AH および ESP ヘッダの組み合わせ
- IKE (Internet Key Exchange) による鍵交換，あるいは各接続ごとの手
動による鍵設定
- 事前共有鍵と公開鍵証明書による IKE 証明。証明書の認証は，RSA 署名
モード，RSA 暗号モード，DSS 署名モードによってサポートされます。
- SysMan アプリケーションによる IPsec ポリシーの構成

IPsec の構成と IPsec に関する問題の解決については『ネットワーク管理ガイ
ド：接続編』を参照してください。

3.1.3.4 アドレス解決プロトコル (ARP)

アドレス解決プロトコル (ARP) は，32 ビットの IPv4 アドレスを，Ethernet
および FDDI リンクで使用される 48 ビットのハードウェア・アドレスに
マップします。シリアル・ライン・インタフェース・プロトコル (SLIP)，
Point-to-Point プロトコル (PPP) はハードウェア・アドレスを持たないた
め，ARP はこれらのプロトコルに対してはアドレス変換を行いません。

ARP は，ローカル・エリア・ネットワーク上で IPv4 アドレスを動的にト
レースしてハードウェア・アドレスを見つけます。このトレースの結果は
マップと呼ばれます。マップされた情報は，マッピング・テーブルに格納
されます。TCP/IP は，ARP を使用してマッピング・テーブルの情報を
収集し配布します。

マッピング・テーブルはカーネルによって保守されるため，ユーザやアプリ
ケーションは ARP を直接使用することはできません。アプリケーションが
インターネット・パケットをインタフェース・ドライバの 1 つに送信する場
合，送信されたドライバは適切なアドレス・マッピングを要求します。マッ
ピングがマッピング・テーブルにない場合には，ARP ブロードキャスト・パ
ケットが，要求しているインタフェース・ドライバを通じてローカル・エリ
ア・ネットワーク上の各ホストに送信されます。

ARP をサポートしている任意のホストが ARP 要求パケットを受け取ると，
要求側システムの IPv4 およびハードウェア・アドレスを確認し，必要があればマッピング・テーブルを更新します。受信側ホストの IPv4 アドレスが要

求されているアドレスと一致しない場合、このホストは要求されているパケットを無視します。IPv4 アドレスが一致する場合、受信側ホストは、要求側システムに応答パケットを送信します。要求側システムは、新しいマッピングを格納し、次の IPv4 パケットの送信に使用します。

多くの他のプロトコルとは異なり、ARP パケットには決まった形式のヘッダはありません。ARP のメッセージは、さまざまなネットワーク環境で使用できるように設計されています。

3.1.3.5 インターネット制御メッセージ・プロトコル (ICMP)

インターネット制御メッセージ・プロトコル (ICMP) は、すべての IPv4 実装に必要となります。ICMP は、IPv4 のエラー・メッセージおよび制御メッセージを処理します。

ICMP の機能は、次のとおりです。

- 送信先が存在し、ルーティングできるかどうか確認する。
- データグラム・ヘッダでパラメータに関する問題を通知する。
- クロックを同期化し、通過時間を算出する。
- インターネット・アドレスとサブネット・マスクを取得する。
- トランスポート・レベルのルーティング情報を提供する。
- ルーティング情報を更新する。

ICMP は、通信環境の問題に関するフィードバックを行いますが、IPv4 の信頼性を高めるものではありません。すなわち、ICMP は、IPv4 パケットを正しく引き渡すことを保証するものではありません。また、IPv4 パケットが引き渡されなかった場合や正しく引き渡されなかった場合に、ICMP メッセージをソース・ホストに戻すことを保証するものでもありません。

ICMP メッセージは、次のような場合に送信されます。

- パケットが送信先に到達できない場合
- ゲートウェイ・ホストが、パケットを転送するだけのバッファ容量を持たない場合
- より良いルートでデータを送信するように、ゲートウェイがホストに指示できる場合

3.2 ネットワーク・インタフェース層

Tru64 UNIX は、次のネットワークへのインタフェースを提供します。

- ATM (Asynchronous Transfer Mode)
- Ethernet
- Fast Ethernet
- Gigabit Ethernet
- FDDI (Fiber Distributed Data Interface)
- トークン・リング
- Serial Line IP および Compressed Serial Line IP
- Point-to-Point Protocol
- Multiple Adapters
- リンク・アグリゲーション
- NetRAIN

3.2.1 ATM

Tru64 UNIX は、155.5 Mb/s の ATM (Asynchronous Transfer Mode) ネットワークで PCI マシンをサポートします。また、622 Mb の ATM ネットワークをサポートするための PCI アダプタもあります。ATM ネットワークは、従来のパケット・スイッチ・ネットワークとは異なり、音声、映像、データなどの異なる種類のトラフィックを同時に転送できる、コネクション・ベースの、高速度、セル・スイッチ・ネットワークです。ATM は、閉ざされたレイテンシと専用の帯域幅を必要とするトラフィックに対して予測サービスを提供します。また、ATM はデータリンク層から物理インタフェースを切り離すため、1 MB/s ~ 10 GB/s の非常に多様な物理インタフェースで同じセルおよびパケット・フォーマットが使用できます。

Tru64 UNIX の ATM の実装は、パーマネント・バーチャル・サーキット・サポート、ポイント・ツー・ポイント接続のための ATM Forum UNI 3.0 および 3.1 シグナリングを通したスイッチ・バーチャル・サーキット・サポート、動的なネットワーク・アドレス登録のための ATM Forum ILMI (Interim Local Management Interface)、クラシカル IP (RFC 1577, RFC 1483 および RFC 1626 で定義)、および、ATM 経由での LAN エミュレー

タ (ATM Forum Version 1 標準で定義) で構成されます。ATM の詳細については、『*Asynchronous Transfer Mode*』および『ネットワーク管理ガイド：接続編』を参照してください。

3.2.2 Ethernet ネットワーク

Tru64 UNIX は、10 MB/s の Ethernet ネットワークをサポートします。

物理レベルおよび IP レベルにおいて、Tru64 UNIX は、最大 10 MB/s、MTU (最大転送ユニット) 1500 バイトで Ethernet ネットワークをサポートします。

IP レベルでの省略時の MTU は、最大 10 MB/s で 1500 バイトです。この値は、`ifconfig` コマンドを使用して減少させることができます。

Ethernet 標準に準拠するよう、Tru64 UNIX は常に最小パケット・サイズ 60 バイトを保証します。

3.2.3 Fast Ethernet ネットワーク

Tru64 UNIX は、100 MB/s の Fast Ethernet (IEEE 802.3 100Base-TX) ネットワークを全二重および半二重でサポートします。

物理レベルおよび IP レベルでの MTU サイズは、通常の 10 MB/s Ethernet の場合と同じです。

3.2.4 Gigabit Ethernet ネットワーク

Tru64 UNIX は、1000 MB/s の Gigabit Ethernet (IEEE 802.3z 1000Base-T) ネットワークをサポートします。また、同期式/非同期式の Pause Frame Flow 制御 (X-on/X-off)、および Jumbo フレームの互換性をサポートします。

物理レベルおよび IP レベルにおける MTU のサイズは、標準の Ethernet と同じですが、この値は、`ifconfig` コマンドを使用して変更することができます。

3.2.5 FDDI ネットワーク

Tru64 UNIX は、すべての Alpha ハードウェア・プラットフォームで、RFC 1042、RFC 1188 に準拠する 100 MB/s FDDI ネットワークをサポートします。

Tru64 UNIX は、物理レベルにおいて、最大 100MB/s、MTU (最大転送ユニット) 4500 バイトで FDDI ネットワークをサポートします。IP レベルでは、最大 100MB/s で MTU 4352 バイトです。

IP レベルでの省略時の MTU は、最大 100 MB/秒 で 4352 バイトです。この値は、`ifconfig` コマンドを使用して減少させることができます。

3.2.6 トークン・リング・ネットワーク

Tru64 UNIX は、RFC 1042 に準拠した 4 MB/s および 16 MB/s のトークン・リング・ネットワークおよびソース・ルーティングをサポートします。

Tru64 UNIX は、物理レベルにおいて、4 MB/s で MTU (最大転送ユニット) が最大 4472 バイト、16 MB/s で最大 17800 バイトのトークン・リングをサポートします。IP レベルでは、4 MB/s の MTU が最大 4092 バイト、16 MB/s で最大 8188 バイトです。

IP レベルにおける省略時の MTU は、4 MB/s でも 16 MB/s でも常に 4092 ですが、この値は、`ifconfig` コマンドを使用して変更することができます。

3.2.7 シリアル・ライン IP (SLIP) および 圧縮シリアル・ライン IP (CSLIP)

Tru64 UNIX はシリアル・ラインに対して IP を完全にサポートしているので、ファイルあるいは NFS マウントしたファイル・システムを電話回線を通して転送することができます。CSLIP `slattach` コマンドでヘッダを圧縮して、転送効率を向上できます。

SLIP/CSLIP のコードは、OSF/1 V1.0 のものです。ただし、OSF/1 は CSLIP の機能にアクセスするための方法を提供していないので、HP は `slattach` コマンドを修正して、CSLIP へアクセスする機能を追加しています。

3.2.8 PPP (Point-to-Point Protocol)

PPP (Point-to-Point protocol) は、シリアル・ポイント・ツー・ポイント・リンクでの データグラム転送の手段を提供します。SLIP とは違い、PPP は、標準カプセル化機能、異なるネットワーク層プロトコルでの同時多重通信、HDLC フレーム・チェック・シーケンスによるエラー検出、非 8 ビット透過電話機および交換機のための HDLC エスケープ機能、IP アドレスのダイナミック・ネゴシエーション機能をサポートします。

また、SLIP が `clist tty` ドライバのみをサポートするのに対して、PPP は、LAN 経由のリモート・ログインはもちろん、STREAMS `clist` と STREAMS ベース `tty` ドライバの 両方をサポートします。

PPP コードは、パブリック・ドメインのものを使用しており、このページの脚注の著作権情報に示された団体が貢献した内容も含まれています¹。PPP コードのセクションの一部は、RSA Data Security 社の MD5 Message-Digest Algorithm に基づいています。

Tru64 UNIX は、PPP Version 2.3.1 をサポートします (RFC 1661)。

PPP の詳細については、『ネットワーク管理ガイド：接続編』、`pppd(8)`、`pppstats(8)` および `chat(8)` を参照してください。

3.2.9 多重アダプタのサポート

Tru64 UNIX では、1 つのシステムが同じサブネットに、複数のアクティブなネットワーク・アダプタを持つことができます。たとえば、IP アドレス 192.24.156.20 で `tu0` を構成し、IP アドレス 192.24.156.21 で `tu1` を構成して、両方が同じネットマスクを持つといった場合です。

接続確立時には、カーネルは接続数が最小のインタフェースを選択します。この接続バランシング効果により、サブネットごとにネットワーク・アダプタが 1 つあるシステムよりもスループットが大きくなります。多重アダプタ・サポートの構成については、『ネットワーク管理ガイド：接続編』を参照してください。

3.2.10 リンク・アグリゲーション

リンク・アグリゲーション (トランキングともいう) を使用すると、管理者は複数の物理 Ethernet ネットワーク・インタフェース・カード (NIC) を結合し、単一の論理リンクを作成することができます (上位層のソフトウェアには、このリンク・アグリゲーション・グループが 1 つの論理インタフェースに見えます)。この単一の論理リンクによるデータのトラフィックは、インタフェースが 1 つの場合よりも高速です。これは、リンク・アグリゲーション・グループを構成しているすべての物理ポートにトラフィックが分散されるためです。

¹ © 1993 The Australian National University. © 1989 Carnegie Mellon University. © 1991 gregory M. Christy. Copyright (c) 1989 Regents of the University of California. © 1990 RSA Data Security, Inc.

リンク・アグリゲーションを使用すると、次の機能を利用できます。

- ネットワーク帯域幅の拡大

拡大の程度は、リンク・アグリゲーション・グループに追加されるポート (NIC) の数やタイプに応じて変わります。

- 耐障害性

リンク・アグリゲーション・グループ内の 1 つのポートに障害が発生した場合、ソフトウェアがその障害を検出し、トラフィックの経路が他の利用可能なポートに変更されます。

- 負荷分散

トラフィックが、リンク・アグリゲーション・グループのすべてのポートに分散されます。

リンク・アグリゲーション・グループ仮想インタフェースは、サーバ間およびサーバとスイッチ間のポイント・ツー・ポイント接続に使用できます。

リンク・アグリゲーションの構成については、『ネットワーク管理ガイド：接続編』、`lag(7)`、および `lagconfig(8)` を参照してください。

3.2.11 NetRAIN

NetRAIN (Redundant Array of Independent Network adapters) は、ネットワーク接続の物理的損失を検出すると、動作状態にあるネットワーク・インタフェースにネットワーク・トラフィックを自動的に切り換えます。

NetRAIN 仮想インタフェースは、同じ LAN セグメント上の複数のインタフェースを 1 つのインタフェースとして構成します。物理インタフェースのうち、常に 1 つがアクティブで、その他はアイドル状態です。アイドル状態のインタフェースも含め、すべてのインタフェースが絶えず監視され、それぞれのインタフェース上をトラフィックが流れることが保証されます。

アクティブ・インタフェースがネットワーク接続に失敗した場合、またはネットワーク接続を失った場合、NetRAIN はネットワーク・トラフィックを次に利用可能な動作中のインタフェースに切り換えます。前に使用したインタフェースのコンテキスト (たとえば、ハードウェア・アドレスやマルチキャスト・アドレス) はすべて維持されます。実際のフェールオーバー時間は、ネットワーク構成と運用に応じて調整可能です。

Tru64 UNIX は、Ethernet、FDDI、および ATM のコントローラをサポートします。

NetRAIN 構成についての詳細は、『ネットワーク管理ガイド』、nr(7) および ifconfig(8) リファレンス・ページを参照してください。

3.2.12 仮想ローカル・エリア・ネットワーク

仮想ローカル・エリア・ネットワーク (VLAN) は、物理ネットワーク上に複数の論理的なネットワークを構築するための機能です。VLAN は、タグ・フレームと呼ばれる特別な Ethernet フレームに含まれる VLAN ID によって認識されます。このタグ付きのフォーマットは、IEEE 802.1q 標準で定義されています。

各 VLAN は、それぞれが物理的に別のネットワークであるかのように機能します。ブロードキャスト・トラフィックを含め、特定の VLAN で発生したすべてのトラフィックは、その VLAN 内の他のリンクのみへ転送されます。詳細は、Version 5.1B の『リリース・ノート』、vlan(7) および vlanconfig(8) リファレンス・ページを参照してください。

3.3 アプリケーション・プログラミング・インタフェース

ネットワーク・プログラミング環境には、アプリケーション、カーネル、およびドライバの開発者が、ネットワーク・アプリケーションを作成したり、ネットワーク・プロトコルを実装するための、プログラミング・インタフェースが含まれます。さらに、アプリケーションがデータの処理および伝送に必要とする、カーネル・レベルのリソースも含まれます。これには、ライブラリ、データ構造体、ヘッダ・ファイル、トランスポート・プロトコルなどがあります。

次のセクションでは、サポートされているアプリケーションのプログラミング・インタフェースについて、簡単に説明しています。

- X/Open トランスポート・インタフェース (XTI/TLI)
- BSD 4.3 ソケット
- System V Release 4.0 STREAMS フレームワーク
- データ・リンク・インタフェース (DLI)
- データ・リンク・プロバイダ・インタフェース (DLPI)

- 拡張可能 SNMP (eSNMP)
- Basic Sockets API for IPv6
- RSVP API (RAPI)

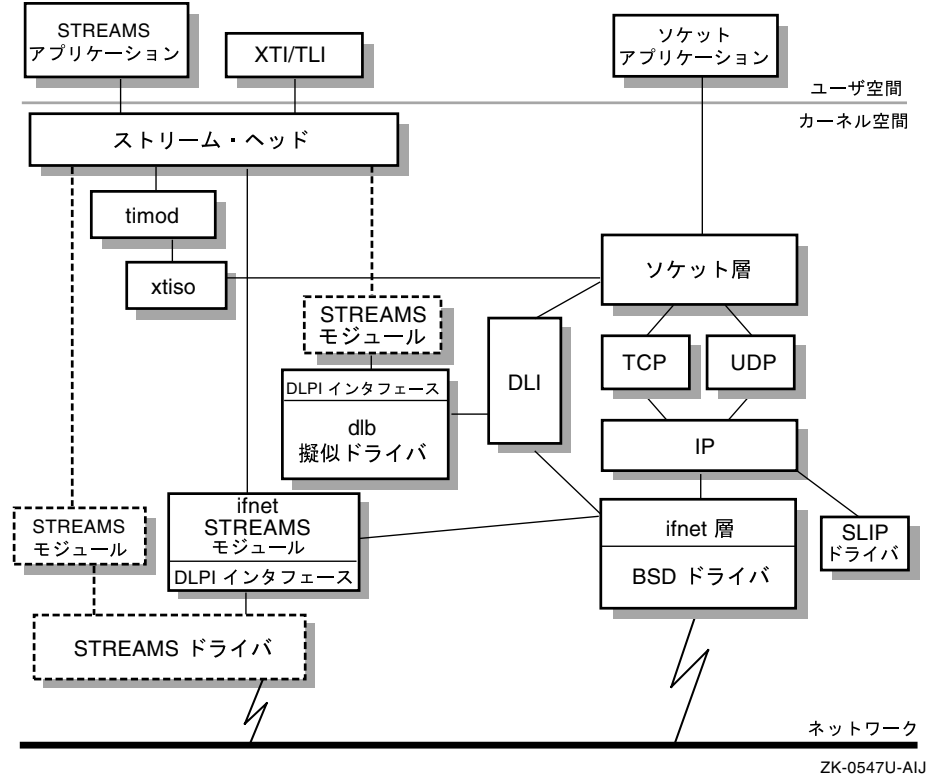
これらのネットワーク・プログラミング環境についての詳細は、『ネットワーク・プログラミング・ガイド』を参照してください。

3.3.1 X/Open トランスポート・インタフェース (XTI)

X/Open トランスポート・インタフェース (XTI) は、トランスポート・プロバイダに依存しない、トランスポート層アプリケーション・インタフェースを定義します。つまり、XTI に合わせて作成されたアプリケーションは、伝送制御プロトコル (TCP) やユーザ・データグラム・プロトコル (UDP) などの、さまざまなトランスポート・プロバイダで実行できます。どのトランスポート・プロバイダを使用するかはアプリケーションが指定します。

図 3-2 は、XTI と、STREAMS フレームワークおよびソケット・フレームワークとのやりとりを示しています。

図 3-2: XTI, STREAMS, およびソケット間の相互関係



アプリケーションによって指定されたトランスポート・プロバイダに応じて、次のいずれかのパスをデータが流れます。

- STREAMSベースのトランスポート・プロバイダが指定された場合
データは、STREAMS 上で実行するように作成されたアプリケーションの場合と同じ経路をたどります。データは、最初にストリーム・ヘッドを通り、次にアプリケーションが STREAMS 上にプッシュした任意のモジュールを通して、最後に STREAMS ドライバに引き渡されます。STREAMS ドライバはネットワークとデータのやりとりをします。Tru64 UNIX では、STREAMS ベースのトランスポート・プロバイダを提供していません。
- ソケット・ベースのトランスポート・プロバイダ (TCP または UDP) が指定された場合
データは timod および xtiso を通じて引き渡されます。該当するソケット層ルーチンが呼び出され、データはインターネット・プロトコル

および `ifnet` 層を通じて BSD ベースのドライバに引き渡されます。
BSD ベースのドライバはネットワークとデータのやりとりをします。

3.3.2 ソケット

ソケットは、業界標準プログラミング・インタフェースです。Tru64 UNIX では、ソケットは、TCP、UDP、IP、ARP、ICMP、SLIP などのインターネット・プロトコルへのインタフェースです。Tru64 UNIX は、4.3BSD (省略時の設定)、4.4BSD、XNS5.0、XNS4.0、および POSIX 1003.1g Draft 6.6 インタフェースをサポートします。

ソケット・フレームワークは、一連のシステム・コール、ライブラリ・コール、ヘッダ・ファイル、およびデータ構造体で構成されています。アプリケーションは、ソケット・システム・コールを通じて、ネットワーク・プロトコルにアクセスできます。また、ソケット・ライブラリ・コールを使用して、ネットワーク情報を操作することもできます。たとえば、`getservent` ライブラリ・コールはサービス名をサービス番号にマップし、`htonl` ライブラリ・コールは着信データのバイト順をローカル・システムのアーキテクチャに適したものに変換します。

ソケットの場合、ユーザ空間のアプリケーションは、適切なソケット・システム・コールにデータを引き渡し、次に、ソケット・システム・コールが、そのデータをネットワーク層に引き渡します。最後に、ネットワーク層が `ifnet` 層を通して、BSDドライバにデータを引き渡します。BSDドライバはネットワークとのデータのやりとりを行います。

ソケットについての詳細は、『ネットワーク・プログラミング・ガイド』、『X/Open CAE Specification, Networking Services (XNS), Issue 5』、『X/Open CAE Specification, Networking Services, Issue 4 (XNS4.0); Protocol Independent Interfaces (POSIX 1003.1g Draft 6.6, Section 5)』および `netintro(7)` リファレンス・ページを参照してください。

3.3.3 STREAMS

STREAMS フレームワークはソケットの代替機能を提供します。この STREAMS インタフェースは、ネットワーク・プロトコルからデバイス・ドライバまであらゆるものを実装するのに使用される、システム・コール、カーネル・ルーチン、およびカーネル・ユーティリティから構成されています。ユーザ空間のアプリケーションは、`open`、`close`、`putmsg`、

getmsg, および ioctl などのシステム・コールを使用して, STREAMS フレームワークのカーネル部分にアクセスします。

Tru64 UNIX は, OSF/1 V1.2 からコードをベースに System V Release 4.0 STREAMS をサポートしており, STREAMS tty インタフェースのサポートを提供しています (Tru64 UNIX は, CLIST あるいは Berkeley ベースの tty インタフェースも引き続きサポートします)。STREAMS についての詳細は, 『ネットワーク・プログラミング・ガイド』を参照してください。

3.3.4 ソケットと STREAMS の相互作用

Tru64 UNIX では, BSD ベースの TCP/IP を使用するプログラムが STREAMS ベースのドライバにアクセスできるよう, ifnet STREAMS モジュールを提供します。このモジュールにより, STREAMS ベースのプロトコル・スタックを使用するプログラムが BSD ベースのドライバにアクセスできるように, Data Link Bridge (DLB) 擬似ドライバが提供されます。

3.3.5 データ・リンク・プロバイダ・インタフェース (DLPI)

DLPI は, OSI 参照モデルのデータ・リンク層にマップするカーネル・レベルのインタフェースです。DLPI を使用すると, ユーザはデータ・リンク・プロバイダの特性について特別な知識を持つ必要がなくなるため, 特定の通信媒体に依存することなくこれらの特性を実現できます。DLPI は, データ・リンク・サービスを使用あるいは提供する STREAMS モジュールを主なターゲットとしているカーネル・レベルのインタフェースです。

Tru64 UNIX では, DLPI インタフェースの一部分のみがサポートされます。詳細については, 『ネットワーク・プログラミング・ガイド』を参照してください。

3.3.6 拡張可能 SNMP インタフェース (eSNMP)

Tru64 UNIX は, Tru64 UNIX ホスト・システム上でユーザ作成のプログラムを分散 SNMP エージェントとして機能させることができるアプリケーション層 API である eSNMP (extensible SNMP) をサポートします。

ユーザ・プログラム (サブエージェント) は, SNMP MIB オブジェクトを eSNMP マスタ・エージェント /usr/sbin/snmpd に動的に登録することができ, これによって, これらのオブジェクトに対する SNMP プロトコル操作を処理することができます。

協調関係にあるプロセス間での MIB オブジェクトの分散は、SNMP アプリケーションに対して透過的です。アプリケーションは、SNMP RFC で指定されている標準トランスポート・エンドポイントを使用して、すべての MIB オブジェクトにアクセスできます。eSNMP API (libesnmp.so) は、snmpd との通信に RFC 2741 (AgentX) を使用します。下位互換性を維持するためのこの変更により、サブエージェントが RFC 2741 に準拠した SNMP エージェントとやりとりすることが可能になっています。

eSNMP 開発ツールは、オプションのプログラミング・サブセット (PMR) に含まれています。詳細については、『ネットワーク・プログラミング・ガイド』を参照してください。

3.4 ネットワーク管理ソフトウェア

Tru64 UNIX では、さまざまなネットワーク管理ソフトウェアをサポートします。以降の各項で、これらのソフトウェアについて説明します。

3.4.1 ネットワーク・コマンドおよびユーティリティ

Tru64 UNIX は、OSF/1 V1.2 に含まれている `finger`、`ftp`、`rdump`、`rdist`、`routed`、`gated`、`telnet`、`tftp` などのネットワーク・コマンドをサポートしています。`bootpd` 機能が、`joind` デーモンに組み込まれおり、DHCP あるいは BOOTP プロトコルを使用してクライアントを構成することができます。

また、Tru64 UNIX は、`inetd` デーモンによって起動される次に示す ONC (Open Network Computing) Version4.2 のユーティリティ・プログラムをサポートします。

- `rusers` および `rusersd`
- `rwall` および `rwalld`
- `spray` および `rsprayd`
- `rup` および `rstatd`
- `rquotad` および `pcnfsd`

3.4.2 Ethernet パケット・フィルタおよびパケット・フィルタ・アプリケーション

Ethernet パケット・フィルタは、ユーザ定義のネットワーク・プロトコルを含むパケットの受信および送信機能とともに、ネットワーク・パケット・ヘッダのデマルチプレクサ機能を提供するソフトウェア・ドライバ・インタフェースです。フィルタ固有のネットワーク・プロトコルに対して使用する場合、このパケット・フィルタは Ethernet モニタとしても機能します。

注意

Ethernet パケット・フィルタはオプションのカーネル・サブシステムですので、実行中のカーネルにパケット・フィルタが構成されていない場合は、パケット・フィルタ・カーネル・ルーチンの呼び出しを行うアプリケーション・プログラムは正しく動作しません。詳細については、`packetfilter(7)`を参照してください。

Tru64 UNIX は、次のパケット・フィルタ・アプリケーションをサポートします。

- **Reverse Address Resolution Protocol デーモン** (`/usr/sbin/rarpd`)
`rarpd` デーモンは、`/etc/ethers` ファイルを使用して、マシンの Ethernet アドレスを、そのマシンの Internet Protocol (IP) アドレスにマップします。このデーモンは、IP アドレスを遠隔のパソコン・クライアントに供給できます。
- **TCP/IP トレースおよび監視ユーティリティ** (`/usr/sbin/tcpdump`)

Tru64 UNIXは、バージョン2.2.1 の `tcpdump` ユーティリティをサポートします。このバージョンの `tcpdump` は、バークレイ・パケット・フィルタ (BPF) 言語を使用します。

`tcpdump`ユーティリティは、Ethernet、トークン・リング、Memory Channel、FDDI、および ATM ネットワークにおける TCP/IP ネットワーク・アクティビティのデバッグおよび分析に使用されます。このユーティリティは、その他のプロトコル群およびインタフェース (PPP、NFS およびループバックを含む) もサポートします。`tcpdump` ユーティリティには、カリフォルニア大学ローレンス・バークレイ研究所およびその関係者によって開発されたソフトウェアが含まれています。

- ログ・ファイル・ユーティリティ (/usr/sbin/tcplice)

tcplice ユーティリティは、tcpdump トレース・ログ・ファイルを扱います。ログ・ファイル全体をダンプするのではなく、tcpdump ログ・ファイルの一部分を選択して、選択した部分のトレースのみを表示することもできます。

- NFS 監視ユーティリティ (/usr/sbin/nfswatch)

Tru64 UNIX は、パーデュー大学の nfswatch バージョン 4.1 をサポートします。nfswatch ユーティリティは curses をベースとしており、LAN 上の NFS サーバとクライアント間の NFS トラフィックを表示します。

- NFS ログ・ファイル要約ユーティリティ (/usr/sbin/nfslogsum)

nfslogsum ユーティリティは、nfswatch によって生成されたログ・ファイルをトラフィック分析サマリに要約します。この出力は、ネットワークのトラブルシューティングに有用です。

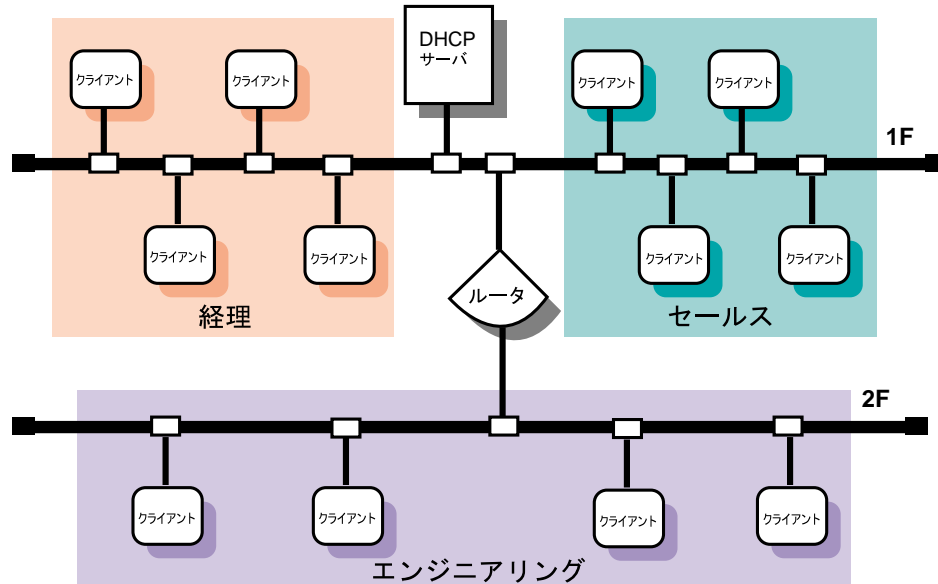
3.4.3 DHCP

Tru64 UNIX は、DHCP サーバが IP アドレスをクライアントに割り当てるためのクライアント/サーバ・フレームワークである JOIN Systems 社の JOIN Server Version 4.1 をベースにした DHCP (Dynamic Host Configuration Protocol) をサポートします。DHCP サーバは、BIND サーバの名前あるいは省略時のルータなどの構成情報をクライアントに提供します。

新しいシステムを初めてブートすると、DHCP サーバがユニークな IP アドレスをシステムに割り当てます。たとえば、そのシステムを同一 LAN 上の別の場所 (異なるサブネット) に移動した場合、DHCP サーバは、そのサブネットに合った新しい IP アドレスを、必要に応じて最初のブート時に、システムに割り当てます。

このように、DHCP (図 3-3) を使用すると IP アドレスの割り当てに関して頭を悩ます必要がなくなります。DHCP は IP アドレスを自動的に割り当てるため、システム管理者の介入が不要になります。

図 3-3: DHCP 構成



ZK-1146U-AIJ

DHCP の詳細については、『ネットワーク管理ガイド：接続編』および dhcp(7)を参照してください。

3.4.4 インターネット・ブート・プロトコル・デーモン (bootpd)

インターネット・ブート・プロトコル・デーモン bootpd は、RFC 951、RFC 1542 および RFC 2132 で定義されているようにインターネット・ブート・プロトコル (BOOTP) サーバをインプリメントします。

BOOTP は拡張可能な UDP/IP ベースのプロトコルで、ユーザの介入なしで、ブート・ホストが自分自身を動的に構成することを可能にします。BOOTP プロトコルは、IP アドレスをホストに割り当て、サーバからダウンロード可能なブート・プログラムを含むファイルを利用可能にし、そのサーバのアドレスを提供し、さらに存在する場合はインターネット・ゲートウェイのアドレスを提供します。

DHCP と同じように、BOOTP プロトコルはネットワーク・アドレスの集中管理を可能にします。

joind デーモンは、bootpd の機能を提供し、DHCP サービスも提供します。

3.4.5 SNMP エージェント

Tru64 UNIX の SNMP エージェントは、ネットワーク管理に利用する次のような多数の管理情報にアクセスすることができます。

- インターネット MIB からの、システム情報、ネットワーク・インタフェース情報、アドレス解決 (ARP) 情報、ルーティング (IP および ICMP) 情報、トランスポート層 (TCP および UDP) 情報 (RFC 1213)
- ホスト・リソース MIB からの、プロセス、ファイル・システム、メモリ、接続デバイスなどのホスト情報 (RFC 1514)
- FDDI MIB からの、FDDI インタフェース情報 (RFC 1285)
- Taken Ring MIB からの、Taken Ring インタフェース情報 (RFC 1748)
- Ethernet ライク・インタフェース MIB からの、Ethernet カウンタ (RFC 1398)
- 3.4.6 項 で説明する IP ルーティング関連 MIB の情報
- サーバ・システム MIB からの、ファームウェア、メモリ、デバイス構成、および環境モニタリングなどのサーバ情報
- システム管理 MIB で構成される限界値、アラーム、およびアクション情報
- Insight Manager で使用する HP 独自の MIB からのシステム情報

eSNMP エージェントにより、Tru64 UNIX ホスト上で管理情報ベース (MIB) の動的な追加が可能です。

マスタ・エージェント、API、ベース・オペレーティング・システムの MIB サポートは、すべて標準のネットワーク・サブセット (CLINET) に含まれています。

eSNMP 開発者ツールは、オプションのプログラミング・サブセット (PGMR) に含まれています。

3.4.6 gated デーモン

gated デーモンを使用すると、複数のネットワーク・インタフェースを持つホストを RIP、OSPF、EGP、BGP などの種々の IP プルブトコルを扱う IP ルータとして機能させることができます。Tru64 UNIX は、次の項

目をサポートする，NextHop Technologies 社の GateD Release 3.5 gated デーモンをサポートします。

- RIP Version 1 (RFC 1058)
ホスト・ルートの適切な包含，ポイズン・リバーズのない水平分割，上品なシャットダウンなどについて規定しています。
- RIP Version 2 (RFC 1388)
IP マルチキャストの使用，クラスレス・ルーティングのサポート，ネクスト・ホップの使用などを規定しています。
- OSPF Version 2 (RFC 1583)
ローカル・ワイア IP マルチキャスト・サポート，MIB サポート (RFC 1253)，およびリコンフィギュレーションを使用します。
- ルーティング・テーブル MIB (RFC 1354)
- EGP 2 (RFC 904)
MILNET の最適化を行った，この仕様の完全な実装
- BGP Versions 2 および 3 (RFC 1163 および RFC 1267)
BGP MIB (RFC 1269)，AS パス・パターン・マッチング (RFC 1164)，OSPF/BGP インターアクション (RFC 1403) の完全な実装
- BGP Version 4 (RFC 1654)
- ルーティング・ソケット・サポートによる可変サブネット・マスクと改善されたカーネル・ルーティング・テーブルの同期化
- ルーティング・テーブルの拡張
BSD 4.3 Reno radix ツリーをベースに，gated はポリシに基いたフィルタ・ルーティングを実装します。これにより，ネットワーク管理者は，個々のプロトコル，ソースおよびデスティネーション・オートノマス・システム，ソースおよびデスティネーション・インタフェース，前のホップ・ルータ，特定のデスティネーション・アドレスでルーティング情報などのインポートおよびエクスポートを制御できます。

ネットワーク管理者は，柔軟なマスキング機能によってインポートされるルーティング情報のそれぞれの組み合わせごとに優先レベルを指定できます。優先レベルを指定すると，gated は，プロトコルとは無関係にどのルートを使用するかを決定できます。

- MIB プロトコルのサポート (“Get Object” サポートのみ)
 - OSPF V2 MIB (RFC 1253)
 - EGP-MIB (RFC 1213)
 - BGP V3 MIB (RFC 1269)

gated デーモンについての詳細は、`gated(8)`、`gated.conf(4)`、`gated.control(4)`、`gated.proto(4)`、および `gated_intro(7)` リファレンス・ページを参照してください。

3.4.7 screend デーモン

screend デーモンはゲートウェイ・スクリーン機能とともに使用され、システムが IP ゲートウェイとして動作している場合に IP パケットを転送するかあるいは拒否するかを決定します。

ゲートウェイ・パケット・スクリーン機能によって、システム管理者は、ゲートウェイとして動作する Tru64 UNIX システムで、どのパケットを転送するかあるいは転送を拒否するかを制御することができます。このため、ゲートウェイ・パケット・スクリーン機能をネットワーク・セキュリティ・ポリシーの一部として使用することができます。

この機能は、カーネル常駐メカニズムとユーザ・レベルのデーモン `/usr/sbin/screend` で構成されます。パケットを転送する用意ができたら、カーネル・メカニズムはそのパケットのヘッダをデーモンへ送ります。screend デーモンはそのヘッダをチェックし、`/etc/screend.conf` 構成ファイルで定義されているルールに従って、パケットを転送するか拒否するかをカーネルに知らせます。

不適当な構成あるいは潜在的なセキュリティ上の問題をネットワーク管理者が見つけることができるように、これらの決定のすべてあるいは一部を記録することもできます。

3.4.8 UUCP

Tru64 UNIX は、HoneyDanBerバージョンの UUCP (UNIX-to-UNIX Copy Program) をサポートします。UUCP は UNIX システム間の通信をサポートするための一連のプログラムです。UUCP は、2 つの UNIX システム間でバッチによるエラーのないファイル転送およびリモート・コマンドの実行を行うことができます。UUCP は、低速で低コストの通信リンクを通して電子

メール、ネットワーク・ニュース、パブリック・ドメイン・ソフトウェアを転送する場合に最も一般的に使用されています。

UUCP は、2 つのシステム間の直接接続のみをサポートします。電子ニュースおよび電子メールの配信は、第三者による転送に依存しています。ネットワークに接続されたサイトのほとんどは、メールおよびニュースの配信のために別のサイトにファイルを中継することに積極的です。UUCP ネットワークの継続的運用は、遠距離の直通ダイヤル・ネットワークおよび閑散時の長距離レートに依存しています。UUCP についての詳細は、`uucp_setup(8)`を参照してください。

3.4.9 ローカル・エリア・トランスポート (LAT)

LAT (Local Area Transport) は、ローカル・エリア・ネットワーク (LAN) 上のホスト・コンピュータと、ターミナル・サーバに接続されている端末、PC、プリンタ、モデム、およびその他のデバイスとの通信をサポートするプロトコルです。LAT ソフトウェアは、ホストがサービス・ノードとして動作するのに必要な機能を提供するため、サーバ・ユーザが接続要求を行うことができます。また、サーバ・ポートへの接続開始、アプリケーション・ポートの指定、リモート・デバイスへのアクセスをホスト・アプリケーションが行うこともできます。LAT ドライバは STREAMS ベースで、4000 ユーザまでサポートしています。理論上の上限は 5000 ユーザです。

Tru64 UNIX では、LAT は SVR4 および BSD 型の両方のターミナル・デバイスをサポートしています。必須のシリアル・ターミナル・デバイスとシリアル・ターミナル・オプションは、LAT として同じ BSD `tty` ネームスペースを共有します。このことは、ユーザがシリアル・ラインにスペシャル・ファイルを割り当てた場合、これらのスペシャル・ファイルが原因で、構成できる BSD LAT デバイスの数が少なくなることを意味します。

LAT についての詳細は、`lat_intro(7)` および『ネットワーク管理ガイド：接続編』を参照してください。

3.4.10 ネットワーク・インタフェースの監視

`niffconfig` コマンドは、1 つあるいは複数のネットワーク・インタフェースの接続の喪失を監視するために必要な処理を行います。このコマンドで、インタフェースがどの程度早期に、接続喪失の疑いあるいは喪失を宣言するかを指定するためのタイミング・パラメータを扱うことができます。

インタフェースの監視を開始すると、カーネル TMT (トラフィック監視スレッド) が監視対象のインタフェースの接続を確認し、必要に応じて、ネットワーク・インタフェース障害デーモン (niffd) に知らせます。このデーモンは、動作していないと分類されたネットワーク・インタフェースに対してトラフィックを生成します。niffd デーモンの目的は、インタフェース・パケット・カウンタを増加させて、そのインタフェースが正しく機能していることを示すことです。

詳細は、『ネットワーク管理ガイド：接続編』および `niffconfig(8)`、`niffd(8)`、`nifftmt(7)` リファレンス・ページを参照してください。

3.5 ネーミング・サービス

Tru64 UNIX は、次の分散ネーミング・サービスをサポートします。

- ドメイン・ネーム・サービス (DNS)
- ネットワーク情報サービス (NIS)。以前、イエロー・ページと呼ばれていたサービスです。

`/usr/lib/libc.so` のライブラリ・ルーチンを使用すると、DNS、NIS、およびローカルな `/etc` ファイルに対して透過的にアクセスできます。ネーミング・サービス構成ファイル `/etc/svc.conf` により、特定のデータベースに対して、どのネーミング・サービスをどの順序で照会するかを指示します。

Tru64 UNIX では、NIS 分散環境を DNS 分散環境に変更したり、両方のサービスを同じ環境で実行したりすることができます。DNS および NIS の両ソース・ファイルはともに `/etc` 形式のファイルであることが可能なため、BSD (Berkeley Software Distribution) のソース・エリアは、シンボリック・リンクを使用して両方のサービスで共用することができます。

3.5.1 ドメイン・ネーム・サービス

ドメイン・ネーム・サービス (DNS) は、未知のホスト名とインターネット・プロトコル (IP) アドレスを検索するためのメカニズムです。

Tru64 UNIX における DNS の実装は、Internet Software Consortium によってサポートされている BIND (Berkeley Internet Name Domain) サービスをベースにしています。BIND サービスは、クライアント・システムが DNS サーバからホスト名とアドレスを取得できるというクライアント/サーバ・モデルになっています。

この階層型ネーミング・システムでは、`named` デーモンによって保守されるローカルの名前解決データベースを使用して、名前解決ルーチンがインターネット上の名前とアドレスの解決を行うことができます。ホストが要求した名前がローカルデータベースにない場合は、名前解決ルーチン、またはローカルの名前付きデーモンがリモートの DNS ネーム・サーバに対して照会を行います。

Tru64 UNIX は BIND Version 8.2.2 (RFC 1034 および RFC 1035 に基づいた) をサポートしており、以下の機能が含まれています。

- マスタ・サーバによるスレーブ・サーバへの DNS リソース・レコード変更の通知と情報の動的な更新 (RFC 1996 および RFC 2136)
- IPv6 クライアントまたは Microsoft Windows クライアントがネットワークに追加された場合の DNS リソース・レコードの動的更新 (RFC 2136)
- 柔軟な、分類されたロギング・システム
- 照会、ゾーン転送、更新などに対する IP アドレス・ベースのアクセス制御と認証

DNS を使用して、ローカルの `/etc/host` ファイルあるいは NIS で提供されるホスト・テーブルのマッピング情報を置き換える、あるいは補足することができます。他のすべての分散データベース・アプリケーションに対して NIS を使用すべきではありません。

DNS 環境、DNS の計画と構成、DNS の管理の詳細については、『ネットワーク管理ガイド：サービス編』、『*BIND Configuration File Guide*』および `bind_intro(7)` リファレンス・ページを参照してください。

3.5.2 ネットワーク情報サービス (NIS)

NIS (Network Information Service) は、関係するホストが共通なシステムおよびネットワーク・ファイルへのアクセスを共用するための分散ネーミング・サービスです。NIS では、システム管理者はこれらの共用ファイルを 1 つのシステムで管理できます。

NIS は、ゲートウェイがインターネットから NIS プロトコルへの外部アクセスを許可しない、安全な環境で使用してください。

3.6 タイム・サービス

Tru64 UNIX は、次のタイム・サービスをサポートします。

- ネットワーク・タイム・プロトコル (NTP)
- 時刻同期化プロトコル (TSP)

NTP は極めて精度の高いクロックを基準にするため、TSP より正確な時刻を提供します。一方 TSP は、ネットワーク・ホスト時刻の平均値を基準にして時刻を同期化します。ご使用のシステムがインターネット上になく高精度のタイム・サーバへアクセスしない場合には、TSP を使用するのが適切です。それ以外の場合は、NTP を使用するのが適切です。

3.6.1 ネットワーク・タイム・プロトコル (NTP)

NTP (Network Time Protocol) は、インターネットのような広域通信網およびローカル・エリア・ネットワークの両方のホストに対して、正確で信頼性の高い、同期化された時刻を提供します。特に NTP は極めて精度の高いクロックをトレースして同期化することが可能で、時刻が不正確なクロックを基準にして同期化すること避けることができます。

NTP を実行している各ホストは、現在の時刻の推定値を照会するデータグラムの交換を定期的に繰り返します。パケットが往復する時間を調べることによって、ホストは一方の遅延を推定することができます (往復時間は両方向でほぼ同じという前提です)。一方向の遅延時間を測定し、NTP パケットで返されるタイムスタンプを調べることにより、自身のクロックと照会したホストのクロックとの時間の誤差を計算します。

ホストは一定の期間に複数回リモート・ホストを照会し、複数のサンプル結果をデジタル・フィルタリング・アルゴリズムに送ります。このアルゴリズムは、ひとつのサンプルによって得られる以上に高精度な、遅延時間、クロック・オフセット、クロックの安定性に関する推定値を導き出します。

NTP メッセージには時刻ソースの精度と信頼性についての情報も含まれています。NTP ホストは、公的機関が送信している時報信号の受信機などの精度の高い時刻ソースに直接接続されています。このようなホストは、第 1 層サーバと呼ばれます。他のすべての NTP ホストは、時刻を同期するために参照したホストの層番号より 1 つ大きな番号を層番号として使用します。たとえば、第 1 層サーバに同期を取るホストは、第 2 層ホストになります。層の決定は自動的に行われ、ホストの層はその接続によって変更することがあります。

NTP を使用するホストは、どのホストが最も正確と思われる時刻を提供しているかをさまざまな情報から判断します。この情報には、デジタル・フィルタリング・アルゴリズムの出力、および照会するホストの層番号が含まれます。他のホストとの通信によって、NTP ホストはクロック時刻の正確でないホストを検出し、長期間通信ができないホストに対しても同期を取ることが可能です。

実際に NTP は、数千マイルに及ぶ広域通信網上でも数十ミリ秒単位でのクロックの同期化を行うことができます。

Tru64 UNIX は、NTP Version 4 をサポートしています。

NTP プロトコルは、RFC 1305 に記述されています。詳細は、『ネットワーク・プログラミング・ガイド』、`ntp_intro(7)`、および `ntp_conf(4)` を参照してください。

3.6.2 時刻同期化プロトコル (TSP)

TSP (Time Synchronization Protocol) は、`/usr/sbin/timed` デーモンによって使用されるプロトコルです。単純なアプリケーションによって、Ethernet などのブロードキャスト・ネットワーク上の TSP サーバは、周期的に TSP パケットを送信します。ネットワークのホストは、TSP を実行しているホストのうちの 1 つをマスタ・ホストとして選択します。

マスタ・ホストは、以降のプロトコルに関する処理を、障害が発生して新しいマスタ・ホストが選択されるまでの間制御します。マスタ・ホストは他のホストから時刻を集計し、報告されたすべての時刻の平均値を計算した後、マスタ・ホスト自身のクロックをこの平均値に設定し、他のホストのクロックをこの平均値に同期させます。

TSP は、すべての関連ホストの時刻を迅速に同期化します。ただし TSP は、正確と思われるソースまで時刻の追跡を行いませんので、系統的なエラーを修正することはできません。クロックに大きな誤差があったり、ホストの時間設定が正しく行われなかった場合、マスタによって計算され分配された平均時刻に対して大きな影響を及ぼす可能性があります。

ファイル・システム

この章では、Tru64 UNIX オペレーティング・システムで利用できるさまざまなファイル・システムについて説明しています。最初に概要を説明した後 (4.1 節)、次のファイル・システムについて説明します。

- 仮想ファイル・システム (4.2 節)
- Advanced File System (4.3 節)
- UNIX ファイル・システム (4.4 節)
- クラスタ・ファイル・システム (4.5 節)
- ネットワーク・ファイル・システム (NFS) (4.6 節)
- CD-ROM ファイル・システム (4.7 節)
- DVD ファイル・システム (4.8 節)
- メモリ・ファイル・システム (4.9 節)
- /proc ファイル・システム (4.10 節)
- File-on-File マウント・ファイル・システム (4.11 節)
- ファイル記述子ファイル・システム (4.12 節)

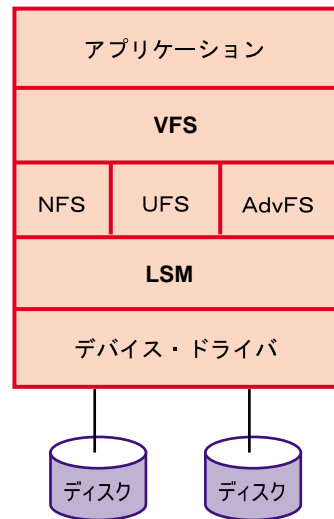
4.1 概要

Tru64 UNIX で利用できるファイル・システムは、すべて仮想ファイル・システム (VFS) 層を通じてアクセスされ、仮想メモリ UBC (Unified Buffer Cache) に統合されます。

ファイル・システムは、ローカルのファイル・システムまたはネットワークに接続されたファイル・システムと対話する、仮想ファイル・システム層によって処理されます。Tru64 UNIX の標準のファイル・システムは Advanced File System (AdvFS) ですが、従来の UNIX ファイル・システムも利用できます。ネットワークに接続されたファイル・システムあるいはローカルのファイル・システムは、LSM と結び付き、さらにデバイス・ドライバ

および実際のストレージ・デバイスに作用します。図 4-1 にファイル・システム，LSM，およびストレージ・デバイスの相互関係を示します。

図 4-1: ファイル・システム



ZK-1577U-AIJ

クラスタ環境でサポートされるファイル・システムについては、『クラスタ概要』を参照してください。

4.2 仮想ファイル・システム

VFS (Virtual File System) は、Berkeley 4.3 Reno 仮想ファイル・システムをベースにしています。VFS は、ファイル・システム層から抽象化された一貫したインタフェースを、ユーザおよびアプリケーションに対して提供することにより、属しているファイル・システムに関係なく各ファイルに対して共通にアクセスすることを可能にしています。この結果、異なるファイル・システムにまたがるファイルに対して、透過的なアクセスが可能になります。

vnode と呼ばれる構造体には、マウントされているファイル・システムの各ファイルについての情報が含まれています。vnode は inode に似た機能を持っており、ファイル・システム固有のノードを包含しています。たとえば、ファイルに対する読み取り要求あるいは書き込み要求が行われた場合、vnodeはそのファイル・システムに適したシステム・コールをポイントします。読み取り要求が AdvFS 上のファイルに対して行われた場合は、advfs_read がポイントされ、要求が UFS 上のファイルに対して行

われた場合は、`ufs_read` がポイントされ、要求が NFS マウントされたファイル・システム上のファイルに対して行われた場合は、`nfs_read` がポイントされます。

Tru64 UNIX の VFS の実装は、拡張ファイル属性 (XFA) をサポートし、任意のアプリケーションがファイルに対して XFA を割り当てることができるようになっています。AdvFS と UFS の両方で XFA をサポートします。XFA についての詳細は、`setproplist(2)` リファレンス・ページを参照してください。

4.3 Advanced File System

Advanced File System (AdvFS) は、柔軟性、互換性、データ可用性、高性能、単純化されたシステム管理機能などを提供する、Tru64 UNIX の標準のルート・ファイル・システムです。このログ・ベース・ファイル・システムは、16 テラバイトに達する長さのファイルとファイルセットを処理します。

AdvFS の構成は、従来の UNIX ファイル・システムとは異なります。AdvFS では、ディレクトリ層とは独立して物理ストレージ層が管理されます。システム管理者は、ファイル・システムのアンマウントやオペレーティング・システムの停止を行わずにストレージの追加、および削除を実行することができます。このため、構成の計画は、より簡単かつ柔軟になります。

ユーザ側から見た場合、AdvFS は他の UNIX ファイル・システムと同じように動作します。エンド・ユーザは、`mkdir` コマンドを使用して新しいディレクトリを作成し、`cd` コマンドを使用してディレクトリを移動し、`ls` コマンドを使用してディレクトリの内容を一覧表示することができます。AdvFS の論理構造、クォータ制御、バックアップ機能は、伝統的なファイル・システムの設計をベースにしています。AdvFS には、`mkfdmn`、`mkfset` などのファイル作成コマンド、`vdump`、`vrestore` などのファイルセットのバックアップ/リストア・コマンドなど、いくつかの独自のシステム管理ユーティリティが用意されています。AdvFS のコマンドおよびユーティリティについては、『*AdvFS 管理ガイド*』を参照してください。

システム管理者は、システムをオフライン状態にしなくても、バックアップ、ファイル・システムの再構成、ファイル・システムの調整を行うことができます。エンド・ユーザは、システム管理者の助けを受けなくても、不用意に削除してしまったファイルをあらかじめ定義されたゴミ箱ディレクトリやクローン・ファイルセットから取り出すことができます。

オペレーティング・システムとは別のライセンスが必要な AdvFS Utilities により、その他のファイル管理機能と Web ベースの GUI が使用可能になり、システム管理を単純化できます。共通デスクトップ環境 (CDE) で実行されるこの GUI によって、メニュー、グラフィカル表示、およびオンライン・ヘルプが使用できるようになり、AdvFS の操作が容易になります。さらにこの GUI によって、システムの状態情報の表示が可能になります。

AdvFS Utilities は、マルチボリューム・ファイル・システムをサポートしています。マルチボリューム・ファイル・システムを使用すると、ファイル・レベルのストライピング (データを複数のボリュームに分散させること) が可能で、入出力を多用するアプリケーションのファイル転送速度が改善されます。ボリューム・レベルのストライピングが可能な Logical Storage Manager (LSM) を、AdvFS 構成の中に組み込むこともできます。

4.4 UNIX ファイル・システム

UNIX File System (UFS) はローカル・ファイル・システムです。以前は UFS が標準のファイル・システムとして位置付けられていましたが、AdvFS が標準のファイル・システムとなった現在でも、AdvFS の代替ファイル・システムとして提供されています。システム・ディスクにおいて、あるいは AdvFS の先進の機能を必要としない場合には、UFS に慣れ親しんだ多くの管理者は UFS の方を選択します。

UNIX ファイル・システム (UFS) は、Berkeley 4.3 Tahoe リリースと互換性があります。UFS では、パス名の構成要素に 255 バイト、絶対パス名に最大 1023 バイトが使用できます。Tru64 UNIX で提供している UFS では、2 GB を超えるファイル・サイズをサポートします。

UFS は、大規模ストレージ・デバイスで raw デバイスと同じスピードのシーケンシャルな読み取り/書き込みアクセスを可能にする、ファイル・ブロック・クラスタリングをサポートします。UFS は、STREAMS に対するファイル・オン・ファイル・マウントをサポートします (4.11 節を参照)。

4.5 クラスタ・ファイル・システム

TruCluster Server クラスタでは、仮想ファイル・システムとしてクラスタ・ファイル・システム (CFS) が物理ファイル・システムの上に存在しており、これにより、クラスタ単位でマウント済みのファイル・システムへのアクセスを可能としています。

CFS により、すべてのファイルがすべてのクラスタ・メンバから見えるようになり、各メンバからそれらのファイルにアクセスできるようになります。ファイルが格納されているデバイスがすべてのクラスタ・メンバに接続されているデバイス上であるか、あるいは 1 つのメンバ専用のデバイス上に関わらず、各メンバ・システムからは同じようにファイルが見えます。CFS は、クラスタ・メンバ間でキャッシュの一貫性を保つことにより、クラスタ内にマウントされているファイル・システムがすべてのメンバから常に同じように見えることを保証しています。

CFS から見ると、各ファイル・システムまたは AdvFS ドメインは、1 つのクラスタ・メンバによりクラスタ全体にサービスされているように見えます。どのクラスタ・メンバも、クラスタ内の任意のデバイス上のファイル・システムのサービスを行うことができます。クラスタのブート時にマウントされたファイル・システムは、最初にそのファイル・システムにアクセスしたクラスタ・メンバがサービスを行います。したがって、あるクラスタ・メンバ専用のバスに接続されたデバイス上のファイル・システムは、そのメンバがサービスを行うことになります。

クラスタ・ファイル・システムについての情報は、『クラスタ概要』を参照してください。

4.6 NFS

ネットワーク・ファイル・システム (NFS) は、異種のプロセッサ、オペレーティング・システム、あるいはネットワークの混在する環境でファイルを共有するための機能です。NFS を使用すると、リモートのファイル・システムあるいはディレクトリをローカル・システムにマウントし、それらのファイルに対して、ローカル・ファイルと同じように読み取り/書き込みすることができます。

Tru64 UNIX は、NFS Version 3 および NFS Version 2 をサポートします。NFS Version 2 のコードは、Sun Microsystems 社からライセンスを受けている ONC Version 4.2 をベースにしています。NFS Version 3 は Sun Microsystems 社のプロトタイプ・コードをベースにしています。

Tru64 UNIX は NFS Version 3 および Version 2 の両方をサポートするため、NFS クライアントおよびサーバの両方でサポートしている高いバージョンの NFS を使用してファイル・システムがマウントされます。たとえば、NFS Version 3 をサポートする NFS サーバによってファイルがサービス

されている場合、Tru64 UNIX クライアントは NFS Version 3 を使用します。しかし、NFS Version 2 をサポートする NFS サーバによってファイルがサービスされている場合、NFS クライアントは NFS Version 2 を使用します。NFS Version 3 の詳細については、『*NFS Version 3: Design and Implementation*』(USENIX, 1994) を参照してください。

NFS の基本サービスに加えて、Tru64 UNIX は以下の拡張機能をサポートします。

- NFS over TCP
- 一括書き込み (Write-gathering)
- NFS ロッキング
- オートマウント機能
- PC-NFS
- WebNFS

4.6.1 NFS Version 3 の機能

NFS Version 3 は、NFS Version 2 で提供するすべての機能に加えて、次の機能をサポートします。

- 性能の向上
 - NFS Version 2 での書き込み性能を改善し、信頼性の高い非同時書き込みをサポートします。その結果、クライアントのレスポンス、サーバ I/O ローディングの性能が向上します。
 - READDIRPLUS プロシージャをサポートします。これにより、ディレクトリ名とともにファイル・ハンドルおよびファイル属性を返してディレクトリ探索時の LOOKUP 呼び出しを省略できます。
 - 後続の GETATTR プロシージャの呼び出し数を低減するために、すべての操作でサーバがメタデータを返します。
 - クライアントがより効果的にキャッシュを管理できるように、キャッシュ・データの一貫性を弱める機能を提供します。
- セキュリティの改善
 - スーパユーザ許可モードの割り当てに関する NFS Version 2 での問題を解決した ACCESS プロシージャが提供されてます。これによ

り、ファイル・オープン時のアクセス・チェックが可能になり、より優れたアクセス制御リスト (ACL) サポートが可能になりました。

- 可変長文字列のファイル名およびパス名をサポートします。フィルタ名およびパス名の最大長は、PATHCONF プロシージャによってクライアントとサーバの交渉の結果により決定されます。
- 排他的ファイル作成の保証

4.6.2 NFS に対する拡張

NFS Version 3 の機能に加えて、Tru64 UNIX は次の NFS 機能を拡張しています。

- NFS over TCP

従来の NFS は、UDP プロトコル上で実行されます。Tru64 UNIX では、TCP プロトコル上で NFS がサポートされます。詳細については `mount(8)` を参照してください。

- 一括書き込み (Write-gathering)

この機能により、NFS サーバで同一ファイルに対する複数の同時書き込み要求がまとめられ、実際に書き込みを行う回数をできるだけ少なくします。連続する書き込み処理のデータ部分はキャッシュに保管され、単一のメタデータ更新が実行され、それがすべての書き込みに適用されます。一括書き込みが NFS のクラッシュ・リカバリ・デザインに違反しないように、すべてのデータおよび関連するメタデータがディスクに書き込まれるまで、応答はクライアントへ送られません。

この結果、一括書き込みは書き込みスループットを 100% 増加させ、書き込みに関する CPU の負荷を大幅に低減させることが可能で、サーバの処理能力を向上させることができます。

- NFS ロック機能

ファイル領域へのアクセスを制御する `fcntl` システム・コールを使用すると、NFS ロック機能により、NFS 経由でファイル・レコードをロックすることができます。この機能により、NFS で提供するデータベースの共用セグメントを保護することができます。状態デーモン `rpc.statd` は NFS サーバをモニタし、サーバがダウンした場合の NFS ロックを管理します。このため、NFS サーバが回復した時に、アクセスを取り戻したプロセスがロックを再設定することが可能になります。

- オートマウント機能

NFS クライアントでは、`/etc/fstab` ファイルを使用してリモートのファイル・システムをマウントする代わりに、`automount` デーモンと `autofs` デーモンを使用して、ファイル・システムを必要に応じてマウントすることができます。

上記のいずれかのデーモンが実行されているシステムのユーザが、リモート・マウントされたファイルやディレクトリにアクセスするコマンドを呼び出した場合、デーモンは該当するファイル・システムまたはディレクトリをマウントし、ユーザが必要とする間、マウントされた状態を維持します。ファイル・システムまたはディレクトリへのアクセスがないまま一定期間（省略時の設定は 5 分）が経過すると、デーモンはそれらをアンマウントします。

マウントするファイル・システムは、ユーザの環境に合わせてカスタマイズ可能なマッピング・ファイルで指定します。マッピング・ファイルの管理は、ローカルで、またはNIS 経由で行うことができ、この 2 つの方法を組み合わせることもできます。

オートマウントによる NFS マウント・ファイル・システムは、静的なマウントと比べて次のような利点があります。

- NIS マッピングを使用している場合にファイル・システムを他のサーバへ移動しても、ユーザはそのファイルにアクセスするために特別な操作を行う必要はありません。ファイル・システムをマウントする必要があるたびに、デーモンが正しい位置のファイル・システムをマウントします。
- ファイルが読み取り専用で、複数の NFS サーバが所定のファイル・システムをサービスしている場合、デーモンは応答の最も速いサーバにユーザを接続します。また、いずれか 1 つのサーバが有効であれば、マウントがハングすることはありません。
- 一定期間（省略時の設定では 5 分）以上アクセスのない NFS マウント・ファイル・システムはデーモンによってアンマウントされ、システム・リソース、特にメモリを節約します。

`autofs` デーモンには、`automount` デーモンより有利な点があります。前者は、カーネル・デーモンとユーザ空間デーモン間の通信が少なく済み、可用性が高いため、後者より効率が高くなっています。

マウントやアンマウントを行うためには `autofs` が実行されている必要がありますが、`autofs` が強制終了されたり利用できなくなった場合でも、オートマウントされている既存の NFS ファイル・システムは引き続き利用できます。

`automount` デーモンと `autofs` デーモンについての詳細は、『ネットワーク管理ガイド：サービス編』、および `automount(8)`、`autofs(8)`、`autofsmount(8)` の各リファレンス・ページを参照してください。

- PC-NFS

HP が提供する PC-NFS サーバ・デーモン `pcnfsd` により、PC-NFS で PC クライアントを構成し、次のことを行うことができます。

- NFS ファイル・システムのマウント

`pcnfsd` プロトコル V1.0 および V2.0 に準拠した PC-NFS `pcnfsd` デーモンは、PC クライアントが NFS と通信できるように UID および GID を PC クライアントに割り当てます。

`pcnfsd` デーモンは、UNIX へのログイン時に行うのと同じようなパスワードとユーザ名の確認を、サーバ上で PC クライアントに対して行います。認証が成功すると、`pcnfsd` デーモンは、そのユーザ名に対して与えられているのと同じ許可コードを PC クライアントに与えます。`mountd` デーモンと通信し、サーバ上の `/etc/exports` ファイルでその NFS ファイル・システムが PC クライアントに対してエクスポートされていれば、PC クライアントは NFS ファイル・システムをマウントすることができます。

DOS にはファイル許可コードのチェックのためのメカニズムが用意されていないため、PC クライアントは認証サーバを呼び出し、そのユーザのファイル属性に関する信任データをチェックします。この処理は、ファイルのオープンのような、許可コードの確認が必要なファイル・アクセスを PC クライアントが要求した場合に行われます。

- ネットワーク・プリンタへのアクセス

`pcnfsd` デーモンは PC クライアントの認証を行い、クライアントに代わってファイルのスプールおよび出力を行います。

- WebNFS

WebNFS は、ローカル・ファイルにアクセスするのと同じ方法で、インターネット経由でクライアントがファイルにアクセスすることを可能にする NFS プロトコルです。WebNFS は、ファイアウォールを越えて機能することができるパブリック・ファイル・ハンドルを使用します。パブリック・ファイル・ハンドルは、接続の初期化に要する時間の低減にも効果があります。パブリック・ファイル・ハンドルは、WebNFS サーバ上の単一のディレクトリに対応しています。詳細については、[exports\(4\)](#)、[exportfs\(2\)](#)、および [nfs_intro\(4\)](#) リファレンス・ページを参照してください。

4.7 CD-ROM ファイル・システム (CDFS)

CD-ROM ファイル・システム (CDFS) はローカル・ファイル・システムです。初期の CD-ROM フォーマットと互換性のある High Sierra Group 標準あるいは Rock Ridge Interchange Protocol (RRIP) バージョン 1.0 リビジョン 1.09 を使用した CD-ROM フォーマットなど、複数のベンタ間のデータ交換ができるように、Tru64 UNIX は ISO-9660 CDFS 標準をサポートします。

RRIP は、ISO-9660 によって定義されているシステム使用領域を使用して、複数セッション、ファイル名における大文字/小文字の区別、長いファイル名の使用、シンボリック・リンク、デバイス・ノード、深いディレクトリ構造、ファイルの UID/GID と許可コード、POSIX タイムスタンプなどの機能を提供できるよう、ISO-9660 を拡張しています。

加えて、Tru64 UNIX は、X/Open Preliminary Specification (1991) CD-ROM サポート・コンポーネント (XCDR) をサポートします。XCDR を使用すると、定義されたユーティリティおよびシェアード・ライブラリを通して、選択した ISO-9660 属性をユーザが調べることができ、システム管理者が、省略時の CD-ROM ファイルのファイル・プロテクション、ファイル所有者、ファイル名を別の値に置換することもできます。詳細については、[cdfs\(4\)](#) リファレンス・ページを参照してください。

4.8 DVD ファイル・システム

DVD (Digital Versatile Disk) ファイル・システムを使用すると、Optical Storage Technology Association (OSTA) Universal Disk Format (UDF) 仕様でフォーマットされたディスクを読み取ることができます。これらのインタフェースは、ISO/ITEC 13346:1995 標準および ISO 9660:1988 標準に準拠しています。

DVD-ROM のユーザ・データ・セクタには、任意のフォーマットの任意のタイプのデータを記録することができます。ただし、Tru64 UNIX の DVDFS システムでは、OSTA UDF ファイル・フォーマット標準に準拠していなければなりません。さらに、DVD-ROM 標準では、論理セクタ・サイズと論理ブロック (ユーザ・データ・ブロック) サイズが 2048 バイトでなければなりません。

詳細は、`dvdafs(4)` のリファレンス・ページを参照してください。

4.9 メモリ・ファイル・システム(MFS)

メモリ・ファイル・システム (MFS) は、メモリ上で機能する UNIX ファイル・システムと言えます。ファイル構造あるいはデータのディスクへの書き込みは行わないため、MFS の内容は、リブート時、アンマウント時、電源オフ時に消失します。ディスクへのデータ書き込みを行わないため MFS は非常に処理の速いファイル・システムで、一時ファイルの保管、既存ファイルの読み取り専用ファイルとしてのロードには、極めて効果的です。

たとえば、ソフトウェアのビルドなど、失敗した場合には再実行が必要となるような処理を行っている場合、処理中に作成される一時ファイルを保管するには処理速度の点で MFS が非常に適しています。

Tru64 UNIX のファイル・システムは仮想ファイル・システムなので、MFS のサイズは物理的な空きメモリのサイズには制限されません。アクティブにアクセスされていない MFS 上のデータはスワップ領域に移され、システム全体の性能が改善されます。MFS は TruCluster Server 構成ではサポートされません、

詳細については、`newfs(8)`を参照してください。

4.10 /proc ファイル・システム

/proc ファイル・システムは、システム・コール `open` , `close` , `read` , `write` , `lseek` , および `ioctl` によって、プロセスをファイルとしてアクセスあるいは処理できる、ローカル・ファイル・システムです。

/proc ファイル・システムは、VFS の下層に位置付けられる擬似ファイル・システムで、ディスク領域は占有しません。このファイル・システムは、`mount` および `unmount` コマンドでマウント/アンマウントすることも、`/etc/fstab` ファイルにエントリを定義することもできます。

クラスタ・システムでは、各クラスタ・メンバが専用の `/proc` ファイル・システムを持っています。

`/proc` ファイル・システムは、他の実行プロセスを制御するのに適切な許可コードになるよう各プロセスを設定できるため、デバッガで使用するのに最も適しています。デバッガとデバッグされるプロセスの間には、親子の関係が存在する必要はありません。詳細については、`proc(4)` を参照してください。

4.11 File-on-File マウント・ファイル・システム

File-on-File マウント(FFM) ファイル・システムは、通常ファイル、キャラクタ・ファイル、ブロック特殊ファイルを通常ファイルにマウントします。

主に STREAMS ベースのパイプ (あるいは FIFO) のシステム・コール `fattach` および `fdetach` で使用されます。これらのシステム・コールは SVR4 互換です。FFM を使用することにより、通常は対応するファイル・オブジェクトを持たない FIFO に、ファイル・システム領域で名前が与えられます。この結果、FIFO を作成したプロセスには関係のないプロセスから FIFO をアクセスすることができるようになります。

`fattach` システム・コールを通じて FFM を使用する プログラムの他に、ユーザが `mount` コマンドを使用して、1 つの通常ファイルを他のファイルの先頭にマウントすることができます。他のファイルの先頭にファイルをマウントしても、それによってカバーされたファイルの内容は破壊されません。単にカバーされたファイルの名前とマウントしたファイルが関連づけられ、カバーされたファイルの内容は一時的に使用できなくなります。カバーされたファイルへのアクセスは、そのファイルにマウントしたファイルがリブート、`fdetach` 呼び出し、あるいは `umount` コマンドによってアンマウントされれば再び可能となります。

ただし、カバーされたファイルの内容は、`fattach` 呼び出しでそのファイルをオープンするプロセスに対しては、あるいはユーザがそのファイルに対して `mount` コマンドを発行した場合には有効です。

クラスタ・システムでは、FFM をマウントしているクラスタ・メンバは FFM を認識しますが、それ以外のクラスタ・メンバは認識しません。

FFM についての詳細は、`ffm(4)` リファレンス・ページを参照してください。

4.12 ファイル記述子ファイル・システム(FDFS)

ファイル記述子ファイル・システム (FDFS) を使用すると、アプリケーションは UNIX ファイル・システム上のファイルと同じようにプロセスのオープン・ファイル記述子を参照することができます。FDFS は、プロセスのオープン・ファイル記述子をファイル・オブジェクトに割り当てます。FDFS がマウントされている場合、ファイル記述子ファイルのオープンあるいは作成は dup システム・コールの呼び出しと同じ効果があります。

FDFS によって、UNIX スタイルの入出力をサポートするようには作成されていないアプリケーションが、パイプ、名前付きパイプ、および入出力のリダイレクトを使用できます。FDFS は、Tru64 UNIX システムには構成されません。FDFS は、コマンドによってマウントするか、システムの `/etc/fstab` ファイルにエントリとして配置しなければなりません。FDFS の詳しい説明は、`fd(4)` リファレンス・ページを参照してください。



カーネル，シンメトリック・マルチプロセッシング，**NUMA**，仮想メモリおよびデバイスのサポート

この章では，次のトピックについて説明します。

- Tru64 UNIX カーネル (5.1 節)
- シンメトリック・マルチプロセッシング (5.2 節)
- 不均等メモリ・アクセス (NUMA) (5.3 節)
- 仮想メモリ・サブシステム (5.4 節)
- Tru64 UNIX によるデバイスのサポート (5.5 節)

5.1 Tru64 UNIX カーネル

カーネルは，Tru64 UNIX のシステム・リソースを管理します。システムの属性を設定することによって，最大の性能が得られるようにこのカーネルを調整できます。さらに，カーネルのチューニングとデバッグのためのツールによって，これらの属性を検査できます。

5.1.1 カーネルのチューニング

Tru64 UNIX には，カーネルの定義や拡張のためのさまざまなサブシステムが用意されています。カーネル変数は，システム・ブート後のサブシステムの動作の制御や，サブシステム統計情報の追跡に使用されます。

カーネル変数は，ブート時に省略時の値を割り当てられます。システム構成や作業負荷の状況によっては，特に，メモリ/ネットワークを多用するシステムでは一部の属性の省略時の値が適切でなく，これらを変更して性能を最適化しなければならないことがあります。

デバッガを使用して，実行カーネルのカーネル変数の値を直接変更することも可能ですが，カーネル・サブシステム属性を使用してカーネル変数にアクセスすることをお勧めします。詳細は `sys_attrs(5)` を参照してください。

サブシステムの属性は、構成マネージャ・サーバ `cfgmgr` によって管理されます。属性の表示と変更は、`sysconfig` コマンドと `sysconfigdb` コマンド、およびカーネル・チューナ `dxkerneltuner` を使用して行います。場合によっては、`sysconfig` を使用して、システムの動作中に属性を変更することもできます。システムのリブート後も変更を保持したい場合は、`sysconfigdb` を使用して属性を変更します。

詳細は、『システムの構成とチューニング』、`sysconfig(8)`、および `sysconfigdb(8)` リファレンス・ページを参照してください。

5.1.2 拡張されたカーネル・デバッグ機能

`dbx` デバッガは、カーネルの変数とデータ構造のチェック、修正、表示が可能なシンボリック・デバッガです。

`dbx` を使用すると、次の処理を実行できます。

- ストライプ化されたイメージのデバッグ
- メモリ内容のチェック
- カーネル変数の値およびカーネルのデータ構造の値と形式の表示
- マルチ・スレッドのデバッグ
- `-k` オプションによるカーネル・コア・ファイルのデバッグ
- `-remote` オプションによる、実行中のカーネルに対するシリアル回線経由でのブレークポイント・デバッグ

`dbx` のフロントエンドは `kdbx` と呼ばれ、各 `dbx` コマンドをサポートするだけでなく、C ライブラリ API もサポートします。これにより、`dbx -k` または `dbx -remote` だけを使用する場合より簡単に、カーネル・データの抽出とフォーマットを行う C プログラムの作成が可能になります。

`dbx` デバッガは、コマンド行プログラムです。もう 1 つの選択肢である `ladebug` デバッガは、コマンド行インタフェースと GUI の両方が用意されています。

詳細は、『*Kernel Debugging*』、『プログラミング・ガイド』、『*Ladebug Debugger Manual*』、および `ladebug(1)`、`dbx(1)`、`kdbx(8)` の各リファレンス・ページを参照してください。

5-2 カーネル、シンメトリック・マルチプロセッシング、NUMA、仮想メモリおよびデバイスのサポート

5.2 対称型マルチプロセッシング

対称型マルチプロセッシング (シンメトリック・マルチプロセッシング: SMP) は、複数のプロセス (またはスレッド・アプリケーションの複数のスレッド) を複数の CPU 上で同時に実行する機能です。複数のプロセスまたはスレッドを同時実行するため、処理速度が大幅に向上します。また、この機能を利用して CPU カードを追加することにより、システムの計算能力と製品寿命を延ばすことができるため、新たにシステムを購入するよりも対費用効果の高い処理能力の増強を行うことができます。

Tru64 UNIX は、計算サーバ (計算のみを行う、マルチスレッド・アプリケーション専用システム)、および、データ・サーバ (多数のネットワーク・クライアントに対してサービスするファイル・サーバ、DBMS サーバ、TP システム、メール・ルータ) の性能を最適化するように設計された SMP の実装をサポートしています。さらに、オペレーティング・システムは、SMP 環境におけるマルチスレッド・アプリケーションの開発をサポートします。SMP を利用すると、マルチプロセッサ・システムをタイムシェアリング・システムとして使用してもマイナスの効果が発生しません。

Tru64 UNIX の SMP は、次のような機能を使用します。

- シンプル・ロック

保持されたロックがタイムアウトとなって解放されるまで、一定時間ロックを引き延ばす (スピンする) ため、スピン・ロックとも呼ばれます。

- コンプレックス・ロック

ロックが解放されるまでアクセスをブロックする読み/書きロックです。

- ファネリング (絞り込み)

ロックが有益でない場合に稀に使用されます。ファネリングは、指定された CPU で強制的にプロセスを実行します。一般的に従来型の、スレッド対応でなく、マルチプロセッサ・セーフでないプログラムで使用されます。

Tru64 UNIX SMP は、ロックで保護しなければならないようなシステム状態を減らしてできるだけ多くの同時性を確立します。その結果、ロックの必要性とそれを行った場合のオーバーヘッドを減少させます。

Tru64 UNIX は、そのカーネルも含め、複数のプロセスまたは複数のスレッドが複数の CPU 上で同時に実行できるよう、完全に並列化されています。このオペレーティング・システムは、同時処理とそのロック・ストラテジによって、同じカーネル・データ構造の完全性を確保します。複数のプロセスおよび複数のスレッドが、同じカーネル・データ構造にアクセスできますが、オペレーティング・システムは、このアクセスが論理的順序で行われるよう管理します。マルチ・プロセスおよびマルチ・スレッドが、互いのロックを保留および要求することはできません。

Tru64 UNIX SMP はまた、指定した CPU に特定のプロセスを割り当てる機能であるプロセッサ・バインディング、および、スケジューラが、すべての実行可能なプロセスを、利用可能なすべての CPU に分配しようとするロード・バランシングをサポートしています (ロード・バランシングよりもプロセッサ・バインディングが優先されます)。

性能を向上させるために、スケジューラは、それぞれのプロセスをそれらが最後に実行された CPU 上で実行し、CPU のキャッシュに状態情報が残っている場合はそれを利用します。

SMP は、システム・ブート時に次の 5 つのいずれかのモードに構成することができます。

- ユニプロセッシング
- 最適化リアルタイム・プリエンブション
- 最適化 SMP
- 最適化リアルタイム・プリエンブションおよび SMP
- ロック・デバッグ・モード

ユニプロセッシングに設定されている場合、システム・ブート時にはマルチ・スレッドのサポートに必要なロックのみでカーネルが初期化されます。

ロック・デバッグ・モードが設定されている場合、システムは次のように動作します。

- ロック階層と SPL (最小システム優先順位レベル) をチェックする
- デバッグ情報をクラス別に格納し、ロックの統計情報を維持する。
- 各 CPU が持つシンプル・ロックを CPU 固有の配列に記録する。

5-4 カーネル、シンメトリック・マルチプロセッシング、NUMA、仮想メモリおよびデバイスのサポート

- スレッドがスレッド構造体に持っているコンプレックス・ロックをすべて記録する。

dbx デバッガを使用すると、このデバッグ情報にアクセスできます。

さらに、開発環境は、マルチスレッド・アプリケーションの開発をサポートします。dbx, profile, および pixie ユーティリティは、複数のスレッドをサポートし、システムにはスレッドセーフ・ライブラリが含まれています。

Tru64 UNIX の開発環境および Tru64 UNIX がサポートするスレッド・パッケージの詳細については、『プログラミング・ガイド』および『*Guide to POSIX Threads Library*』を参照してください。SMP 構成の詳細については、『システム管理ガイド』および『システムの構成とチューニング』を参照してください。

5.3 不均等メモリ・アクセス (NUMA)

シンメトリック・マルチプロセッサ (SMP) システムには、一般的にすべてのシステム・リソースにリンクするインターコネクト (バスまたはスイッチ) が 1 つあります。つまり、システム内のすべての CPU は、システムのメモリや入出力チャネルにアクセスするときの遅延と帯域幅が同じになります。従来の SMP システムのアーキテクチャの欠点は、システムの CPU の数を増やすと、システム・バスが性能のボトルネックになることです。

このボトルネックへの 1 つの対処方法は、SMP ブロック (それぞれ、限られた数の CPU、メモリ・アレイ、入出力ポートが存在する) からシステムを構築し、ブロック間を接続する第 2 レベルのバスまたはスイッチを追加することです。不均等メモリ・アクセス (NUMA) は、このタイプのシステム・アーキテクチャを説明するときに使う用語です。これは、特定の CPU がメモリおよび入出力リソースにローカルにアクセスする (CPU と同じビルディング・ブロックにある) かあるいはリモートでアクセスする (他のビルディング・ブロックにある) かによって、帯域幅と遅延が異なるためです。

NUMA サポートについての詳細や、完全なプログラム例については、次のサイトのオンライン・ドキュメント・セットに含まれている Web 専用ドキュメント『*NUMA Overview*』を参照してください。

http://www.tru64unix.compaq.com/docs/base_doc/DOCUMENTATION/V51_HTML/NUMA/TITLE.HTM

5.4 仮想メモリ

仮想メモリ・サブシステムは、次の機能を実行します。

- メモリをプロセスに割り当てる。
- システムにあるすべてのページを追跡管理する。
- ページングとスワッピングを使用して、プロセスの実行やファイル・システム I/O のキャッシュに十分なメモリを確保する。

物理メモリの合計量は、システムにインストールされたメモリ・ボードの容量によって決まります。システムはこのメモリをページと呼ばれる 8 KB の単位で分配します。システムは物理メモリのページを次の 3 つのエリアの間で分配します。

- 固定メモリ

メモリは、ブート時には静的に、実行時には動的に固定されます。

ブート時には、オペレーティング・システムと PAL (Privileged Architecture Library) コードが、基本のシステム操作を実行するために、物理メモリの隣接部分を固定します。静的固定メモリは、オペレーティング・システムのデータやテキスト、システム・テーブル、メタデータ・バッファ・キャッシュ (最後にアクセスした UFS および CDFS を一時的に保持する)、および AdvFS (Advanced File System) バッファ・キャッシュのために確保されます。静的固定メモリは、ページングによって再生されることはありません。

- 仮想メモリ

仮想メモリ・サブシステムは、物理メモリの一部を使用して、プロセスが最後にアクセスした匿名メモリ (変更可能な仮想アドレス空間) およびファイルの裏付けがあるメモリ (file backed memory) をキャッシュします。このサブシステムは、競合するプロセス間にメモリを割り当て、すべての物理ページの分配を追跡します。このメモリは、ページングとスワッピングによる再生が可能です。

- ユニファイド・バッファ・キャッシュ

UBC (Unified Buffer Cache) は、物理メモリの一部を使用して、最後にアクセスしたファイル・システム・データをキャッシュします。

UBC には、読み書きのための、また、マップされたファイル領域からのページ・フォールトのための実際のファイル・データおよび AdvFS

メタデータが入っています。オペレーティング・システムとストレージ・サブシステムのための層として機能することにより、UBC はディスク操作の数を減少させることができます。このメモリは、ページングによる再生が可能です。

仮想メモリ・サブシステムと UBC は、固定されていない物理ページをめぐって競合します。ページは、必要に応じて、プロセスと UBC に割り当てられます。メモリの需要が高まると、最も古い (最も前に使用された) ページが仮想メモリ・サブシステムおよび UBC から再生され、スワップ領域に移動されてから再利用されます。さまざまな属性により、仮想メモリ・サブシステムと UBC が利用できるメモリの量およびページ再生の割合が制御されます。

5.4.1 ページの管理と追跡

仮想メモリ・サブシステムは、必要に応じて、物理ページをプロセスと UBC に割り当てます。物理メモリには限界があるので、これらのページは再利用できるよう、定期的に再生されます。

仮想メモリ・サブシステムは、ページ・リストを使用して、すべての物理メモリ・ページの位置と世代を追跡します。すべての物理ページは、いつでも次のリストのいずれかにあります。

- 固定リスト
固定され、再生できないページ。
- フリー・リスト
クリーンで使用されていないページ。このリストのサイズは、ページの再生がいつ発生するかを制御する。
- アクティブ・リスト
仮想メモリ・サブシステムまたは UBC で使用されているページ。
どのページを最初に再生するかを判定するために、ページ・スティーラ・デーモンがアクティブ・リストの中から最も古いページを探し、これらの LRU (least recently used) ページを次のように指定する。
 - 非アクティブ・ページとは、仮想メモリ・サブシステムによって使用されている最も古いページである。
 - UBC の LRU (least recently used) ページとは、UBCによって使用されているページの中で最も古いページである。

Tru64 UNIX の仮想メモリは、NUMA に対応しています。Tru64 UNIX では、RAD ごとにリストと作業用スレッドのセットを管理します。

5.4.2 変更されたページのプリライト

仮想メモリ・サブシステムは、メモリ不足から容易に回復できるよう、メモリ・ページをクリーンな状態に保とうとします。フリー・リストにあるページがすぐに枯渇すると仮想メモリ・サブシステムが予測する場合は、最も前に変更された (ダーティな) 非アクティブ・ページのスワップ領域に、あらかじめ書き込み (プリライト) を行います。さらに、UBC の変更 LRU ページが UBC の合計 LRU ページの 10 % を超える場合、仮想メモリ・サブシステムは、最後に変更された UBC の LRU ページのスワップ領域にプリライトします。

5.4.3 属性を使用したページングとスワッピングの制御

メモリの需要が高まり、フリー・リストの不足が顕著な場合、ページングが始まります。仮想メモリ・サブシステムは、最も古い非アクティブ・ページおよび UBC の LRU ページを判定し、その変更済みページの内容をスワップ領域に移動してから、クリーン・ページをフリー・リストに入れると、そこでそれらが再利用されます。

フリー・ページ・リストが個々のページを再生するだけでは補充できない場合、スワッピングが始まります。スワッピングは、プロセスを一時的に中断し、常駐セット全体をスワップ領域に移動し、そこで大量の物理メモリが解放されます。

ページングとスワッピングが開始、停止するタイミングは、チューニング可能な仮想メモリ・サブシステム内のカーネル属性の値によって異なります。

UBC は仮想メモリ・サブシステムとカーネルに固定されていない物理ページをめぐって競合しているので、UBC に対するメモリの割り当てがファイル・システムの性能やページング/スワッピング・アクティビティに影響することがあります。UBC は動的で、ファイル・システムの要請の変化に対応するために、消費メモリの量が変化します。

省略時には、UBC は最高 100% のメモリを消費することができます。ただし、UBC に割り当てられているメモリの一部は、仮想メモリ・サブシステムから借りているに過ぎません。ページングが始まると、借りている UBC

ページがまず再生されます。UBC に割り当てられるメモリの量は、仮想メモリ・サブシステムのカーネル属性によって制御されます。

5.4.4 ページング操作

メモリの需要が高く、フリー・ページ・リストにあるページ数がページング限界値を下回ると、仮想メモリ・サブシステムがページングを使用してフリー・ページ・リストを補充します。ページ再生コードにより、ページングとスワッピングが制御されます。ページ・アウト・デーモンおよびタスク・スワップ・デーモンは、ページ再生コードの拡張機能です。

ページ再生コードは、ページ・スティーラ・デーモンをアクティブ化し、それによって、UBC のサイズが借用限界値 (省略時の値は 20 %) に到達するまで、UBC が仮想メモリ・サブシステムから借りているページがまず再生されます。再生されたページがダーティな (変更されている) 場合、ページをフリー・ページ・リストに移動する前に、その内容をディスクに書き込む必要があります。UBC ページは通常変更されないの、借りている UBC ページを解放することが、ページを再生する最も速い方法です。

UBC が借りているメモリを解放してもフリー・リストが十分に補充できない場合、ページ・アウトが起こります。ページ・スティーラ・デーモンが最も古い非アクティブ・ページと UBC の LRU ページを再生します。

フリー・ページの数がおお減少し続ける場合、ページングがますます積極的に行われます。フリー・ページ・リストのページ数が 20 ページを下回ると (省略時の値)、リストにあるそれぞれのページについて、ページを再生しなければならなくなります。デッドロックを避けるため、フリー・ページ・リストにあるページ数が 10 ページを下回る場合 (省略時の値)、フリー・ページ・リストが補充されるまで、特権を持つタスクのみがメモリを割り当てられるようになります。これらの限界値は、チューニング可能な属性によって制御されます。

ページ・アウトは、フリー・リストにあるページ数がページング限界値を上回ったときに停止します。個々のページのページングだけでは、フリー・リストを十分に補充できない場合は、大量のメモリを解放するために、スワッピングが使用されます。

5.4.5 スワッピング操作

メモリの需要が高い場合、仮想メモリ・サブシステムはページの再生によってフリー・リストの補充を行えないことがあります。スワッピングは、プロセスを中断することにより物理メモリに対する需要を減少させるので、フリー・リストのページ数を劇的に増加させます。プロセスをスワップ・アウトするために、タスク・スワップがプロセスを中断し、その常駐セットをスワップ領域に書き込み、クリーン・ページをフリー・リストに移動します。スワッピングは、システムの性能に重大な影響を与えることがあります。

アイドル・タスクのスワッピングが始まるのは、フリー・リストにあるページ数が、一定時間、スワッピング限界値 (省略時の値は 74 ページです) を下回ったときです。次にタスク・スワップが、30 秒以上アイドル状態にあるタスクをすべて中断します。

フリー・リストにあるページ数がなお減少し続ける場合、ハード・スワッピングが始まります。タスク・スワップが、優先順位が最も低く、常駐セット・サイズが最も大きいタスクを一度に 1 つずつ中断します。

個々のタスクのスワッピングは、フリー・リストにあるページ数がスワッピング限界値 (省略時の値は 1280) を上回ったときに停止します。

スワッピングは、フリー・リストにあるページ数が一定時間、十分に補修されたときに起こります。タスクのワーキング・セットは、スワップ領域からページ・インされ、実行できるようになります。省略時の設定では、タスクは 1 秒間スワップ・インした後でスワップ・アウトすることができます。

スワッピングの割合を増加させる (ページの再生中、早いタイミングでスワッピングを行う) と、スループットが上昇します。スワップ・アウトされるプロセスが増加すると、実際に実行されるプロセスが減少し、より多くの作業が行われます。スワッピングの割合を増加させると、長い間スリープ状態にあるスレッドをメモリから移動して、メモリが解放されますが、これにより対話の応答時間が長くなります。スワップ・アウト・プロセスが必要な場合は、遅延時間が長くなります。

スワッピングの割合を減少させる (ページ再生中、遅いタイミングでスワッピングを行う) と、対話の応答時間は改善されますが、スループットが犠牲になります。

5.4.6 スワップ領域割り当てモード

スワップ領域を割り当てるには、2種類のモードを使用できます。これらのモードは、仮想メモリ・サブシステムが匿名メモリ (変更可能な仮想アドレス空間) 用にスワップ領域を確保する方法が異なります。匿名メモリは、ファイルの裏付けはないがスワップ領域によって裏付けられるメモリです (たとえば、スタック空間、ヒープ空間、および `malloc` 関数によって割り当てられたメモリ)。いずれのモードにも、性能上の利益はありません。

- 即時モード

このモードは、プロセスが初めて匿名メモリを割り当てるときに、スワップ領域を確保します。即時モードは、省略時のスワップ領域割り当てモードで、イーガー・モードとも呼ばれます。

このモードでは、システムが必要以上に大量のスワップ領域をプロセス用に確保することがあります。しかし、プロセスがスワップ領域を必要とするときには必ず確保することができます。スワップ領域が確保できない場合は、要求プロセスが中断します。

- 延期モード

このモードは、仮想メモリ・サブシステムが変更された仮想ページをスワップ領域に書き込まなければならなくなったときのみ、スワップ領域を確保します。匿名メモリ用のスワップ領域の確保は、それが実際に必要になるまで延期されます。延期モードは、レイジー・モードとも呼ばれます。

このモードでは、即時モードよりもスワップ領域が少なくすみ、また必要なスワップ領域の予約量がより少ないので、システムがより高速に実行されます。しかし、延期モードは前もってスワップ領域を確保しないため、プロセスが必要とするときにスワップ領域が利用できないことがあります、プロセスが非同期的に強制終了されることがあります。

強制終了されるプロセスは、メモリを要求しているプロセスではなく、ほとんどの場合、システム・デーモンなどのアイドル状態のプロセスとなります。

さらに、特定のコマンドまたはアプリケーション用のシステム規模のスワップ領域割り当てモードを、`swapon` コマンドを使用して上書きすることができます。詳細については、`swapon(8)` リファレンス・ページを参照してください。

5.4.7 スワップ・バッファの使用

メモリとディスクの間のデータ移動を容易にするために、仮想メモリ・サブシステムは、同期および非同期スワップ・バッファを使用します。仮想メモリ・サブシステムは、これら 2 種類のバッファを使用して、比較的遅いプロセスであるページ・アウト要求の完了を待つことなく、ページ要求にただちに応えます。

同期スワップ・バッファは、ページ・フォールトのページ、およびスワップ・アウトに使用されます。非同期スワップ・バッファは、非同期ページ・アウトおよび変更ページのプリライトに使用されます。

5.4.8 ユニファイド・バッファ・キャッシュ

Unified Buffer Cache (UBC) は、Tru64 UNIX の仮想メモリ機能です。UBC は、マシンの物理メモリの一部を使用して、最後にアクセスされたファイル・システム・データをキャッシュします。UBC には、通常のファイル・アクティビティでの読み書き、マップされたファイル・セクションからのページ・フォールト、および AdvFS メタデータなどの実際のファイル・データが入ります。

UBC は、仮想メモリ・サブシステムと物理メモリを共有 (競合) しますが、カーネルに固定されたページはこの限りではありません。UBC は、ファイル・システムの要請の変化に動的に対応するために、消費メモリの量が変化します。UBC の詳細については、『システムの構成とチューニング』を参照してください。

5.5 デバイスのサポート

カーネルは、I/O デバイスのホットスワップをサポートします。これにより、ホットスワップ可能な I/O デバイスがオンラインになると、デバイス・ドライバが自動的にフォールト・インする能力が提供されます。

ハードウェア・コードは、新しいデバイスを検出し、デバイス・ドライバがカーネルにないかと判定すると、カーネル関数呼び出しを 1 度行って、自動的にデバイスのドライバをカーネルにロードします。さらに、ホットスワッピングにより、カーネル・サブシステムやドライバをカーネルにあらかじめ構築しなくてすむので柔軟性が向上します。これらは、デバイスに最初にアクセスしたときにフォールト・インすることができます。

5-12 カーネル、シンメトリック・マルチプロセッシング、NUMA、仮想メモリおよびデバイスのサポート

この章では、Tru64 UNIX の開発環境で提供する主な機能について説明します。次のトピックについて説明します。

- Compaq C コンパイラ
このコンパイラは、きわめて効果的なコードを生成して Alpha アーキテクチャの 64 ビット・アドレス空間を十分に活用できるように、標準に準拠しながら高度に最適化されています。(6.1 節)
- C , アセンブリ言語 , Fortran (f77 および f90) , C++ , Pascal , および Ada をサポートするデバッガ (6.2 節)
- ATOM , gprof , prof , および uprofile のプロファイリング・ツール (6.3 節)
- シェアード・ライブラリ (6.4 節)
- 実行時ライブラリ (6.5 節)
- Java 開発キット (JDK) (6.6 節)
- 開発コマンド (6.7 節)
- POSIX Thread , Visual Thread , Thread-Independent Service のサポート (6.8 節)
- Berkley Memory-Mapped File Support (mmap) 機能 (6.9 節)
- リアルタイム・ユーザとプログラミング環境 (6.10 節)
- ネットワーク・プログラミング環境 (6.11 節)

また、Tru64 UNIX は、国際化機能、標準の UNIX 開発ツール (awk , lint , make , prof など) , Ada , C++ , Cobol , Fortran , Pascal プログラムのための実行時ライブラリを提供します。

詳細については、『プログラミング・ガイド』 , 『プログラミング・サポートツール・ガイド』 , 『*Assembly Language Programmer's Guide*』 , および『国際化ソフトウェア・プログラミング・ガイド』を参照してください。

6.1 Compaq C コンパイラ

Tru64 UNIX で提供する Compaq C コンパイラは、Alpha アーキテクチャの 64 ビット・アドレス空間を十分に活用した効果的なコードを生成するための、標準に準拠した、高度に最適化された C コンパイラです。Compaq C コンパイラは、次の C 言語をコンパイルできます。

- 拡張 ANSI および ISO C (省略時の設定 `-std`)
- 厳密な ANSI および ISO C (`-std1`, `-isoc94`)
- K&R C (`-std0`)
- Microsoft C (`-ms`)
- VAX C (`-vaxc`)

Compaq C コンパイラは、次の標準に準拠してコンパイルを行います。

- ANSI および ISO C (`cc -std1`)
- XPG4-UNIX (`c89 -D_XOPEN_SOURCE_EXTENDED`)

標準への準拠についての詳細は、standards(5) リファレンス・ページを参照してください。

Compaq C コンパイラは、以下の言語拡張をサポートします。

- OpenMP 並行分解指示文 (`-omp`, `-check_omp`, `-mp`)
- Microsoft C 構造化例外処理 (`try-except` と `try-finally`) およびスレッド・ローカル・ストレージ
- メモリの消費を抑え、ポータビリティを容易にする 32 ビット・ポインタ (省略時は 64 ビット・ポインタ) (`-taso`, `-xtaso`, `-xtaso_short`; 詳細は `protect_headers_setup(8)` リファレンス・ページを参照)
- 命令シーケンス・インラインを拡張し最適化する、`#pragma` およびイントリンシック関数を使用したユーザ定義アセンブリ言語シーケンス
- データ・アラインメントの制御のための `#pragmas` およびコマンド行オプション (`-assume`, `-misalign`, `-Zp`, `-member_alignment`)

Compaq C コンパイラは、次のような算術演算に関する拡張機能をサポートします。

- IEEE 浮動小数点 (NaN, INF などの例外条件の適切な処理を含む)

- Fast math モード (INF , NaN などの変換され例外処理は行わない)
- long double に対する 4 倍精度 (128-bit) 浮動小数点表現

Tru64 UNIX で提供する Compaq C コンパイラについての詳細は、cc(1) リファレンス・ページを参照してください。

6.2 デバugga

Tru64 UNIX は、次のソース・コード・デバuggaをサポートします。

- dbx
- ladebug

6.2.1 dbx デバugga

dbx デバuggaは、ソース・コード・レベルあるいはマシン・コード・レベルでのプログラムのデバuggaに利用できます。dbx がサポートするインタフェースは、コマンド行インタフェースです。dbx を使用して実行できるタスクは次のとおりです。

- C , Fortran (f77 および f90) , アセンブリ言語 , Pascal で記述されたプログラムのデバugga
- アクティブ・カーネル , コアダンプ , 複数のスレッドを使用したプログラムのデバugga (/proc を使用して実行プロセスに結び付ける) , シェアード・ライブラリのデバugga
- カーネルのコアダンプの分析
- ユーザ・プログラムあるいはカーネルのオンディスク・コピーへのパッチの適用
- マルチプロセス・デバuggaの実行と , fork および exec 呼び出しにまたがったデバuggaの実行

6.2.2 ladebug デバugga

ladebug デバuggaは、GUI と dbx と同じようなコマンド行インタフェースの両方をサポートするオブジェクト指向のソース・レベル・シンボリック・デバuggaです。ladebug を使用して実行できるタスクは次のとおりです。

- 実行プロセスのアタッチおよびデタッチ
- fork および exec による検出とデバugga

- マルチ・プロセスのデバッグ
- マルチスレッド・プログラム (POSIX スレッド・アプリケーションあるいはカーネル・スレッドを使用するカーネル・モジュール) のデバッグ
- C++, C, Fortran 77, Fortran 90, Ada, COBOL あるいは アセンブリ言語で作成された プログラムのデバッグ

ladebugは、C++ 名をデマングルし、C++ の式を理解する完全な C++ デバッガで、インライン関数、テンプレート、C++ 例外処理をサポートします。F77/F90 のサポートには、大文字小文字の区別なし、コモン・ブロック、代替エントリ・ポイント、言語に依存するタイプのプリント、シェープ・アレイの想定が含まれます。

- マシン・レベル・コードのデバッグ
- 実行プログラムあるいはコア・ダンプのデバッグ
- シェアード・オブジェクトのデバッグ
- 境界合わせされていないデータへのアクセスに対する処理
- アクティブなカーネルのローカル/リモート・デバッグと カーネル・クラッシュ・ダンプの解析
- ソース・プログラミング言語の構文を使用した式の評価
- 異なるターゲット・マシンで実行しているプログラムのリモート・デバッグ・サーバによるリモート・デバッグ

ladebug リモート・デバッグ・プロトコルは、プロトコルに準拠したサンプル・リモート・デバッグ・サーバと一緒に使用することができます。

国際化サポートは別のキットで提供されます。国際化バージョンの ladebug デバッガは、入力としてマルチバイト文字を受け入れ、デバッガのグローバル・ロケール設定に従ってローカル言語文字を出力します。ladebug デバッガは、C/C++ の `wchar_t` データ型もサポートします。

6.3 プロファイリング・ツール

Tru64 UNIX は、次のプロファイリング・ツールをサポートします。

- ATOM

各種のユーザ定義プログラム分析ツールを構築できる、柔軟なコード計測インタフェースを提供します。ツールキットには、計測制御ユーティ

リティおよびライブラリが含まれており、手順的なインタフェースによって特別な目的の計測および分析ツールを開発することができます。

ATOM は次のような組み込み計測/分析ツールを提供します。

- hiprof

gprofによって処理される出力を生成する セル・グラフ・プロファイリング・ツール

- third (Third degree)

不正なメモリ・アクセスに対するメモリ・リークおよびメモリ・チェックを発見するツール

- pixie

プログラムの実行可能イメージとシェアード・ライブラリのプロファイルを行う既存の pixie 基本ブロック・プロファイラのスーパーセット。pixie の出力は prof によって分析されます。

Tru64 UNIX は、次のプロファイリング・ツールをサポートします。

- gprof

-pg オプションでコンパイルされたプログラムに対して、gprof は、名前付きのプロシージャが相互に実行した呼び出しの回数と、各々のプロシージャが消費した CPU 時間の量を表示します。hiprof で計測されたプログラムの出力の分析も行います。

- prof

-p オプションでコンパイルされたプログラムに対して、プログラムおよびシェアード・ライブラリで各プロシージャが消費した CPU 時間を PC サンプリング統計値を使用して表示します。pixie で計測されたプログラムあるいは uprofile/kprofile でモニタされたプログラムの出力の分析も行います。

- uprofile/kprofile

アプリケーション・プログラムあるいはカーネル自体の実行中に、Alpha チップの組み込み性能カウンタを使用して CPU におけるさまざまなイベントをサンプリングします。prof で分析可能な CPU サイクル、メモリ/キャッシュの効果などをレポートすることができます。

プロファイリング・ツールについての詳細は、『プログラミング・ガイド』およびそれぞれのリファレンス・ページを参照してください。

6.4 シェアード・ライブラリ

Tru64 UNIX は、動的なロードのための System V API (`dlopen`, `dldclose`, `dlsym`, `dlderror`) に加えて、シェアード・ライブラリのロードとシンボル解決のための System V セマンティクスと互換性のある、動的なシェアード・ライブラリを提供しています。

シェアード・ライブラリを使用すると、ルーチンそのものをプログラムに含めるのではなく、ルーチンのロード方法およびアクセス方法に関する情報のみをプログラムに含めることが可能になるため、システムの性能を向上させ、ディスクおよびメモリの使用量を低減させ、システム管理を単純化することができます。

シェアード・ライブラリは `/usr/shlib` ディレクトリにあります。

Tru64 UNIX の X11R6.3 および Motif の実装では、スタティック・ルーチンおよびシェアード・ルーチンの両方をサポートしています。X11 シェアード・ライブラリは、`/usr/shlib/X11` ディレクトリにあります。

6.4.1 クイックスタート

アドレスが競合する場合、ユニークなアドレスによって、より早くシェアード・ライブラリを開始できるようにクイックスタート機能をサポートしています。各シェアード・ライブラリは、`/usr/shlib/so_locations` ファイルにユニークなアドレスを持っている必要があります。これにより、実行時にシェアード・オブジェクトを再配置する必要がないため、シェアード・ライブラリとリンクしているアプリケーションをより早く実行できます。シェアード・ライブラリを作成すると `ld` ユーティリティが `so_locations` ファイルを読み書きします。

6.4.2 ダイナミック・ローダ

Tru64 UNIX は、System V Release 4.0 互換のローダを使用してシェアード・ライブラリを動的にロードします。このローダは次の拡張機能を提供します。

- 動的にロードされたシェアード・ライブラリへの呼び込み
- シンボル・プリエンブション (preemption) を含む、System V Release 4.0 シンボル解決セマンティクス
- プログラム・ロード時間短縮のためのライブラリのプリリンク

6.4.3 バージョニング

Tru64 UNIX は、シェアード・ライブラリの完全なあるいは部分的な重複をサポートしています。ローダは、通常の探索パスにバージョン文字列を追加したパス名を使用して、通常の探索パスのサブディレクトリで古いバージョンのシェアード・ライブラリを探します。このように、複数バージョンのシェアード・ライブラリを保持し、適切なバージョンのシェアード・ライブラリをアプリケーションにリンクできるので、カーネル・インタフェースあるいはグローバル・データ定義に対する変更のように、通常はバイナリ互換性が無くなるような変更が行われた場合も、アプリケーションには影響を与えません。

たとえば、OSF は Motif 1.2 でいくつかのインタフェースを変更したため、Motif 1.1.3 ライブラリを使用して構築したアプリケーションとの互換性が無くなっています。このため Tru64 UNIX では、バイナリ互換性を維持するために、バージョニング機能を使用して Motif 1.1.3 と Motif 1.2 の両方のシェアード・ライブラリを提供し、アプリケーションが必要に応じて Motif 1.1.3 のシェアード・ライブラリにもアクセスできるようにしています。シェアード・ライブラリのバージョニングについての詳細は、『プログラミング・ガイド』を参照してください。

6.5 実行時ライブラリ

Tru64 UNIX は次の実行時ライブラリ (RTL) をサポートします。これらのライブラリにより、対応する言語のプログラムをシステムにインストールしておかなくても、以前にコンパイルしたプログラムを実行することができます。

- **Compaq C++ 実行時ライブラリ** (`libcxx`, `libcomplex`, `libtask`)
これらのライブラリは、I/O 処理、複雑な計算機能、マルチタスキングなどの Compaq C++ 実行時機能をサポートします。
- **Compaq COBOL 実行時ライブラリ** (`libcob`, `libots2`, `libisamstub`)
これらのライブラリは、I/O 処理、10 進法での計算、COBOL ACCEPT/DISPLAY セマンティクス、STRING/UNSTRING 操作、CALL および CANCEL などの Compaq COBOL 実行時機能をサポートします。
- **Compaq Fortran 実行時ライブラリ** (`libfor`, `libfutil`, `libUfor`)
これらのライブラリは、I/O 処理、イントリンシック関数、データ・フォーマット、データ変換、算術関数、通常のオペレーティング・シス

テム・サービスへの Fortran バインドなどの Compaq Fortran 実行時機能をサポートします。

- Compaq Pascal 実行時ライブラリ (libpas)

このライブラリは、I/O 処理、算術関数、タイムサービスおよびデータ・サービス、ファイル・サービスなどの Compaq Pascal 実行時機能をサポートします。

6.6 Java 開発キット

Java™ 開発キット (JDK) が、Tru64 UNIX のコンポーネントとして組み込まれています。このキットを使用して、Java アプレットやプログラムを開発および実行することができます。

JDK for Tru64 UNIX には JIT (just-in-time コンパイラ) が付属しており、アプリケーションの Java バイトコードおよび実行時呼び出しをネイティブの Alpha マシン・コードに、オンザフライでコンパイルします。この結果、Java インタプリタを使用して実行する場合と比較して、Java アプリケーションの実行が大幅に高速化されます。省略時の設定では、java コマンドを入力すると、JIT が実行されます。

JDK は、Java スレッドをネイティブ (POSIX) スレッドの最上位に実装します。これにより、マルチプロセッサ・マシンを使用している場合は、アプリケーションの中の異なる Java スレッドを、別々のプロセッサで実行することができます。また、これは、POSIX スレッドを使用して実装されたネイティブ・メソッドまたはネイティブ API (たとえば DCE) にリンクされたときに、Java アプリケーションが正しく実行されることを意味します。

詳細については、JDK がインストールされている Tru64 UNIX システムの `/usr/share/doc/lib/java/index.html` を参照してください。

6.7 開発コマンド

Tru64 UNIX は、ソース・コード制御システム `rscs` および `sccs` とともに、`ar`、`as`、`btou`、`cb`、`cc`、`cflow`、`cpp`、`ctags`、`cxref`、`c89`、`dbx`、`dis`、`error`、`file`、`indent`、`ld`、`lex`、`lint`、`loader`、`m4`、`make`、`mig`、`mkstr`、`nm`、`odump`、`pixie`、`ppu`、`prof`、`ranlib`、`size`、`stdump`、`strings`、`strip`、`tsort`、`xstr`、および `yacc` などの開発ツールをサポートします。

開発コマンドの多くは System V , POSIX , XPG4 および XPG4-UNIX 標準によって指定されており , Tru64 UNIX はこれらに完全に準拠しています。

6.8 スレッドのサポート

Tru64 UNIX は , 以前は DECthread と呼ばれた POSIX スレッド・ライブラリ , Visual Thread , Thread-Independent Service を提供します。

6.8.1 POSIX スレッド・ライブラリ

POSIX スレッド・ライブラリは , POSIX 1003.1c-1995 標準マルチスレッド API を実装したものです。POSIX スレッドは , 効率的な 2 レベルのスケジューリング (POSIX プロセス・コンテンション・スコープで , カーネルに密接に統合されており , 計算と I/O の同時性のレベルが常に最大になるよう自動的に維持されます。他のプロセスまたはカーネルにあるハードウェア・イベントまたはスレッドに応じて , リアルタイムにスケジュールする必要があるアプリケーションの場合は , POSIX スレッドがシステムコンテンション・スコープとスケジューリング機能を提供します。

6.8.2 Visual Threads

Visual Threads は , マルチスレッド化されたアプリケーションを分析し , デバッグするための診断ツールです。Visual Threads を使うと , マルチスレッド化に関する一般的な問題を自動的に診断できます。これらの問題には , デッドロック , 共有データの保護 , スレッドの使用方法的エラーなどがあります。

また , このツールを使用して , スレッドに関連するアプリケーションの性能を監視することができ , ボトルネックやロックの細分性の問題の識別を支援します。これは , アプリケーションが明確な症状を示さない場合でも , 問題のある領域を識別するのに使用できる , ユニークなデバッグ・ツールです。

Visual Threads は , POSIX スレッド (DECthreads) を使用するアプリケーション , または Java で書かれた Tru64 UNIX アプリケーションであれば , どれに対しても使用することができます。これはあらゆるサイズのマルチスレッド・アプリケーション用に設計されており , スレッドが 2 つのアプリケーションから 数百のアプリケーションまで処理することができます。

Visual Threads は , Developers' Toolkit の一部としてライセンスされます。

6.8.3 TIS (Thread Independent Services)

Tru64 UNIX は、アプリケーション作成者が、非スレッド・ライブラリおよびアプリケーションに対してスレッド・セーフ・コードを作成できるように TIS (Thread Independent Services) ルーチンをサポートします。スレッドがある場合、これらのルーチンは指定したスレッド・セーフ機能を提供します。スレッドがない場合、これらのルーチンは最小限のオーバーヘッドしか呼び出し元に与えません。シングルスレッドおよびマルチスレッドの両方のアプリケーションをサポートするために、TIS ルーチンは C 実行時ライブラリによって使用されます。

6.9 メモリ・マップ・ファイル・サポート(mmap)

Tru64 UNIX は、Berkeley メモリ・マップ・ファイル・サポート (mmap) 機能をサポートします。この機能により、アプリケーションは、I/O 操作ではなくメモリ操作でデータ・ファイルにアクセスできます。

6.10 リアルタイム・ユーザとプログラミング環境

Tru64 UNIX は、リアルタイム・ユーザ環境およびリアルタイム・プログラミング環境をサポートしています。これらのリアルタイム環境は、オプションのリアルタイム・サブセットとして提供されています。Tru64 UNIX リアルタイム・プログラミング環境は、リアルタイムに関する POSIX 1003.1b-1993 標準に準拠しており、POSIX 環境で移植可能なリアルタイム・アプリケーションを開発/実行することができます。POSIX 1003.1b インタフェースは、リアルタイムおよび非同期 I/O ライブラリ、librt および libaio で提供されています。

カーネル・プリエンプション (preemption) 機能を有効にすると、カーネル・モードで実行しているかユーザ・モードで実行しているかにかかわらず、優先順位の高いプロセスを優先順位の低いプロセスより優先することができます。この完全にプリエンプティブなカーネルによって、プロセス・プリエンプション待ち時間 (Process Preemption Latency: 優先順位の低いプロセスより優先するために必要な総時間) を最小にすることができます。

プリエンプティブなカーネルに加えて、Tru64 UNIX リアルタイム・プログラミング環境では、次の POSIX 1003.1b 機能をサポートします。

- リアルタイム・クロックおよびタイマ
- リアルタイム・キュー・シグナル

- 固定プライオリティ・スケジュール・ポリシ
- リアルタイム・スケジューラ優先レベル
- セマフォのカウント
- 共有メモリ
- プロセス・メモリ・ロック
- 非同期 I/O
- 同期 I/O
- メッセージ引き渡しインタフェース
- リアルタイム・ライブラリのスレッドセーフの実現

リアルタイム・プログラミング環境についての詳細は、『*Guide to Realtime Programming*』を参照してください。リアルタイム・カーネルの構成についての詳細は、『インストレーション・ガイド』を参照してください。

6.11 ネットワーク・プログラミング・インタフェース

ネットワーク・プログラミング環境には、アプリケーション、カーネル、およびドライバの開発者がネットワーク・アプリケーションを作成し、ネットワーク・プロトコルを実装するためのプログラミング・インタフェースが含まれます。さらに、ライブラリ、データ構造体、ヘッダ・ファイル、およびトランスポート・プロトコルなど、アプリケーションがデータの処理と伝送を行うために必要とするカーネル・レベルのリソースも含まれます。

このオペレーティング・システムでは、次のプログラミング・インタフェースがサポートされています。

- X/Open トランスポート・インタフェース (XTI/TLI)
- BSD ソケット
- System V Release 4.0 STREAMS
- データ・リンク・インタフェース (DLI)
- データ・リンク・プロバイダ・インタフェース (DLPI)
- 拡張可能 SNMP (eSNMP)

ネットワーク・プログラミング環境の詳細については、『ネットワーク・プログラミング・ガイド』を参照してください。



セキュリティ

この章では、Tru64 UNIX が提供するセキュリティ機能について説明します。最初にセキュリティ機能の概要を簡単に説明し（7.1 節）、その後、次のトピックについて説明します。

- 識別，認証，承認（7.2 節）
- 任意アクセス制御（7.3 節）
- 監査サブシステム（7.4 節）
- オブジェクトの再使用（7.5 節）
- トラストッド・コンポーネントのための保護環境（7.6 節）
- 完全性（7.7 節）

7.1 概要

システム・セキュリティの基礎になるのは識別，認証，承認です。システムのセキュリティは，システム・リソースにアクセスしようとするローカルおよびリモート・ネットワーク内の他のシステムとの信頼性のレベル，あるいはユーザおよびアプリケーション間のシステムの信頼性のレベルに関するセキュリティ・ポリシーの作成によって構築されます。

システム上のリソースにアクセスしようとするエンティティを完全に信頼するためには，システムがそのエンティティを識別し，そのエンティティがシステムに対して自身を証明する必要があります。エンティティが識別され認証されると，そのエンティティがどのリソースにアクセスできるかを決定するルールが施行されます。

Tru64 UNIX は，識別，認証，および承認サービスを提供するいくつかのセキュリティ・メカニズムをサポートします。また，システムにおけるアクティビティを監視するための監査サブシステムを提供します。

識別，認証，承認に関するすべてのセキュリティ・コンポーネントは，コマンド行から管理できます。また，それらのコンポーネントのほとんどは，SysMan インタフェースでも管理できます。

Tru64 UNIX のセキュリティ機能の詳細は，以下のドキュメントを参照してください。

- 『セキュリティ管理ガイド』
- 『セキュリティ・プログラミング・ガイド』
- 『*Tru64 UNIX* ユーザーズ・ガイド』 (一般ユーザ向けの情報)

7.2 識別，認証，承認

Tru64 UNIX は以下のセキュリティ・メカニズムをサポートし，ローカル・システム・アクセス，ローカル・ネットワーク・アクセス，リモート・ネットワーク・アクセスに対してさまざまなレベルの信頼性でセキュリティ・ドメインを構築するセキュリティ・ポリシーを実現します。

- ローカル・システム・アクセス
 - ベース(BSD) セキュリティ
 - エンハンスド・セキュリティ
- ローカル・ネットワーク・アクセス
 - Secure Shell
 - CDSA (Common Desktop Security Architecture)
 - IPsec
 - SSL (Secure Socket Layer)
 - Single Sign-on (SSO)/Kerberos
 - LDAP (Lightweight Directory Access Protocol)
 - NIS (Network Information Services)
- リモート・ネットワーク・アクセス
 - Secure Shell
 - CDSA
 - IPsec

- SSL
- Single Sign-on (SSO)/Kerberos

各メカニズムで識別および認証に関するそれぞれのルールおよび信頼性のレベルを定義して複数のセキュリティ・メカニズムを実現するように、Tru64 UNIX を構成することができます。適用するセキュリティ・メカニズムの順序は SIA (Security Integration Architecture) が管理します。

セキュリティ・メカニズムおよび SIA についての詳細は、『セキュリティ管理ガイド』を参照してください。

7.3 任意アクセス制御

任意アクセス制御 (DAC: Discretionary Access Control) は、ユーザが作成したリソースをどのように共有するかを制御するための機能を提供します。伝統的な UNIX では、許可ビットがこの機能を提供します。

Tru64 UNIX システムは、オプションの ACL (access control list) によって個々のユーザ・レベルでオブジェクトの保護機能を提供しています。ACL は、UFS、NFS、AdvFS、および CFS ファイル・システムでサポートされます。ACL の管理を単純化するために、コマンド行インタフェースに加えて ACL GUI `dxsetacl` が提供されます。

7.4 監査サブシステム

監査サブシステムは、ファイル・オープン、ファイル作成、ログイン、印刷ジョブの発行などのシステム・イベントを記録します。それぞれのイベントには作成ユーザの audit ID (AUID) がスタンプされ、ユーザはすべてのアクションを追跡することができます。ユーザが監査サブシステムを直接操作することはありません。

Tru64 UNIX では次の監査機能を提供します。

- システム監査を簡素化する GUI, `audit_setup`
- コマンド行インタフェース
- 監査記録をリモート・ホストに送信する機能
- システム・タイプに基づいた単純な構成を可能にするイベント・プロファイリング

- システム・イベント，アプリケーション・イベント，サイト定義イベントのきめ細かい事前選択
- システム・イベント，アプリケーション・イベント，サイト定義イベントのきめ細かい後分析
- オプションの監査ログ自動消去機能
- dxaudit GUI

エンハンスド・セキュリティとともに使用すると，監査機能には，強化識別および認証によるユーザごとの監査属性プロファイルもサポートされます。この監査システムは，SysMan あるいはコマンド行から `audit_setup` ユーティリティを使用することによって設定できます。監査システムの保守は，コマンド行から実行するか，dxaudit GUI を使用します。

7.5 オブジェクト再使用

オブジェクト再使用によって，次のタイプの物理ストレージ (メモリあるいはディスク・スペース) が確実に“クリア”されます。

- 共用オブジェクトに割り当てられた物理ストレージ
- 別なユーザに再割り当てされる前に解放された物理ストレージ

オブジェクト再使用の例としては，ファイルが切り捨てられたあと解放されたディスク領域や，別のユーザによる読み取りのために再割り当てされる前に解放された物理メモリがあります。

7.6 トラストッド・コンポーネントのセキュリティ

Tru64 UNIX は，ハードウェア・メモリ管理機能を使用して，自分自身のためのカーネル・アドレス空間を用意し，実行中のアプリケーション・プロセスのインスタンスごとに別のアドレス空間を用意します。各プロセスは，同じアドレス空間に書き込もうとする場合があります。DAC は，このアドレス空間のプロセス間における共用状態を制御します。省略時の設定は，共用不可です。

管理者は，読み取り専用アドレス空間 (たとえばシェアード・ライブラリ) としてセクションの共用を不可にすることができます。このように，システムのセキュリティ関連のコンポーネント (トラストッド・コンピューティング・ベース，すなわち TCB) は，それらの実行中は保護されます。

Tru64 UNIX は、DAC を使用して、オンディスク・セキュリティ・コンポーネントを保護します。DAC の保護違反が検出されたらシステム・セキュリティ管理者が必要な措置を取れるよう、監視する機能も提供されています。

さらに、セキュリティ・コンポーネントは、明確に定義された、非常に独立性の高いモジュールで構成されています。

Tru64 UNIX は、仕様、ドキュメント、ソース・コード、オブジェクト・コード、ハードウェア、ファームウェア、およびテスト・スイートに対する変更を制御する構成管理システムの下で、設計/開発/保守されています。ソース・コードからの新しいバージョンのセキュリティ・コンポーネントの生成を制御および自動化し、正しいバージョンのソースが新しいバージョンに統合されていることを確認するためのツールも提供されています。

セキュリティ・コンポーネントの生成に使用されるすべてのマテリアルのマスタ・コピーは、未許可の変更あるいは破壊から保護されます。

7.7 完全性

Tru64 UNIX は、ハードウェア、ファームウェア、ソフトウェア・セキュリティ・コンポーネントの操作が正しいかどうかを確認するための機能を提供します。ファームウェアには、電源 ON 時の診断プログラムと、必要に応じて使用可能なより詳細な診断プログラムが含まれています。ファームウェア自体は EEPROM に含まれており、物理的に書き込み保護されています。また、ファームウェアは、オフ・ラインのマスタ・コピーと比較したり、オフ・ラインのマスタ・コピーから再ロードすることができます。その他のハードウェア診断プログラムを実行することもできます。

ファームウェアは、省略時の設定以外のオペレーティング・ソフトウェアをロードするための認証、あるいは、メモリのテストおよび修正のためのコンソール・モニタ特権コマンドの実行のための認証を要求することができます。

オペレーティング・システムをロードした後は、ハードウェアおよびソフトウェアの正しい動作を確認するためにシステム診断プログラムを実行することができます。また、オペレーティング・システムの正しい操作を確認するためのテスト・ツールも使用できます。

セキュリティ・ソフトウェアとデータベースにおける矛盾を検出するために、次の 2 つのツールを自動的に実行させることができます。

- fverify

fverify ユーティリティは、標準入力からサブセット・インベントリ・レコードを読み取り、システム上のファイルの属性が、対応するレコードにリストされている属性と一致することを確認します。ファイルの不足と、ファイル・サイズ、チェックサム、ユーザ ID、グループ ID、許可コード、あるいはファイル・タイプに関する矛盾が報告されます。

- authck ユーティリティ

authck ユーティリティは、認証データベースのすべてのコンポーネントに関して、全体的な構造と内部フィールドの両方の一貫性をチェックし、発見したすべての問題を報告します。

国際化機能

この章では、Tru64 UNIX の国際化機能について説明します。最初に国際化の概要 (8.1 節) を簡単に説明し、以降の節で次のトピックについて説明します。

- サポート言語 (8.2 節)
- `localedef` ユーティリティによるロケールの作成 (8.3 節)
- コードセット変換 (8.4 節)
- Unicode ロケールと dense コード・ロケール (8.5 節)
- Unicode 標準 V3.1 および ISO 10646 規格のサポート (8.6 節)
- 「国際化ソフトウェアの構成」オプション (8.7 節)
- ユーロ通貨記号のサポート (8.8 節)
- `dxim` 入力サーバ (8.9 節)
- 国際化 Curses ライブラリ (8.10 節)
- その他の国際化機能 (8.11 節)

8.1 国際化機能の概要

“国際化” (Internationalization) という用語は、The Open Group によって次のように定義されています。

“異なる言語、地域的な異なる習慣、異なる文字コードセットのそれぞれの要求に適合するための機能を、コンピュータ・プログラム内部に準備しておくこと。”

つまり、国際化プログラムは、ソース・コードの変更をせずに、サポートしているすべてのロケールで実行することができます。ロケールとは、中国、フランスなどの特定の地理的あるいは言語的領域における文化的な慣習に正確に対応するソフトウェア環境のことを示し、その領域で使用されている言語がサポートされます。たとえば、中国語のロケールを設定した場

合、コマンド、システム・メッセージおよびキー入力はすべて中国語文字で行われ、中国語に合った方法で表示されます。

Tru64 UNIX は、既存のアプリケーションをユーザの母国語で操作できるだけでなく、国際化アプリケーションの開発を支援するために WPI (Worldwide Portability Interfaces) に準拠するアプリケーション・インタフェースを完全にサポートする国際化オペレーティング・システムです。OSF からリリースされたコードは HP によって拡張されています。

Tru64 UNIX の国際化サポートは、The Open Group のシステム・インタフェースおよびヘッダ (XSH Issue 5)、curses (XCURSES Issue 4.2)、およびコマンドとユーティリティ (XCU Issue 5) の CAE 仕様に準拠しています。これらの仕様は、POSIX および ISO C 標準の現在のバージョンと整合性が取れています。この準拠により、コマンド、ユーティリティ、ライブラリが国際化されており、対応するメッセージ・カタログがベース・システムで提供されています。

Tru64 UNIX は、中国語文字入力標準 GB18030-2000 に準拠しています

さらに、オペレーティング・システムは X 入力メソッド (XIM) および X 出力メソッド (XOM) をサポートしており、ローカル言語文字の入力、テキスト描画、計測単位設定およびクライアント間通信が可能になっています。これらの機能は、X11R6.3 に従って実装されており、X11R6.4 で規定されるいくつかの問題の修正が含まれています。

Tru64 UNIX は、アジア言語用に 32 ビット `wchar_t` データ・タイプをサポートしています。このサポートにより ISO 10646 標準で定義されているものを含め、さまざまなコードセットの使用が可能になります。

Tru64 UNIX の国際化機能については下記のドキュメントを参照してください。

- 『国際化ソフトウェア・プログラミング・ガイド』 (プログラマ向け情報)
- 『国際化機能ユーザズ・ガイド』
- Tru64 UNIX ワールドワイド言語サポートに関する Web ページ

<http://www.tru64unix.compaq.com/unix/i18n.htm>

8.2 サポート言語

ワールドワイド言語サポート (WLS) サブセットに含まれているほとんどのロケールは、オプションでインストールされます。いくつかについては、表 8-1 で示すとおり、必須ベース・オペレーティング・システムの一部となっています。

ロケール名が UTF-8 で終るロケールは、ファイル・コードと内部処理コードに ISO 10646 および Unicode 標準で定義されている文字エンコーディング (wchar_t エンコーディング) を使用します。その他の UTF-8 でない Unicode ロケールは、ファイル・コードには従来の UNIX およびメーカー独自のコードセットを、内部処理コードには UTF-32 を使用します。これらの Unicode ロケールの一部には、ロケール名に @ucs4 の文字列が含まれています。ただし、それらは @ucs4 が付かないロケールと同じものです。

universal.UTF-8 ロケールもサポートされます (エンド・ユーザ向けというよりアプリケーション用に)。このロケールは UCS (universal character set) で一連の文字をサポートします。コーディング・フォーマットについての詳細は unicode(5) を参照してください。

UTF-8 および Latin-9 (ISO 8859-15) ロケールはユーロ通貨記号をサポートします。

サポートする言語およびロケールの最新情報は l10n_intro(5) リファレンス・ページを参照してください。

表 8-1 に示すのは、オペレーティング・システムでサポートする言語およびその対応ロケールです。

表 8-1: 言語とロケール

言語	ロケール名
カタロニア語	ca_ES.ISO8859-1 ^a
	ca_ES.ISO8859-15
	ca_ES.UTF-8

表 8-1: 言語とロケール (続き)

言語	ロケール名
中国語/簡体字 (PRC)	zh_CN.UTF-8
	zh_CN.dechanzi
	zh_CN.dechanzi@ucs4
	zh_CN.dechanzi@pinyin
	zh_CN.dechanzi@pinyin@ucs4
	zh_CN.dechanzi@radical
	zh_CN.dechanzi@radical@ucs4
	zh_CN.dechanzi@stroke
	zh_CN.dechanzi@stroke@ucs4
	zh_CN.GBK
	zh_CN.GB18030
中国語/繁体字 (香港)	zh_HK.big5
	zh_HK.dechanyu
	zh_HK.dechanyu@ucs4
	zh_HK.dechanzi
	zh_HK.dechanzi@ucs4
	zh_HK.eucTW
中国語/繁体字 (台湾)	zh_HK.eucTW@ucs4
	zh_TW.big5
	zh_TW.big5@chuyin
	zh_TW.big5@radical
	zh_TW.big5@stroke
	zh_TW.dechanyu
	zh_TW.dechanyu@ucs4
	zh_TW.dechanyu@chuyin
	zh_TW.dechanyu@chuyin@ucs4
	zh_TW.dechanyu@radical
	zh_TW.dechanyu@radical@ucs4
	zh_TW.dechanyu@stroke
	zh_TW.dechanyu@stroke@ucs4
	zh_TW.eucTW
	zh_TW.eucTW@ucs4
	zh_TW.eucTW@chuyin
	zh_TW.eucTW@chuyin@ucs4
	zh_TW.eucTW@radical
	zh_TW.eucTW@radical@ucs4
	zh_TW.eucTW@stroke
	zh_TW.eucTW@stroke@ucs4
チェコ語	cs_CZ.ISO8859-2
	cs_CZ.ISO8859-2@ucs4
デンマーク語	da_DK.ISO8859-1 ^a
	da_DK.ISO8859-15
	da_DK.UTF-8
オランダ語	nl_NL.ISO8859-1 ^a
	nl_NL.ISO8859-15
	nl_NL.UTF-8

表 8-1: 言語とロケール (続き)

言語	ロケール名
ベルギー系オランダ語	nl_BE.ISO8859-1 ^a nl_BE.ISO8859-15 nl_BE.UTF-8
アメリカ英語/ASCII	C (POSIX) ^a
アメリカ英語	en_US.ISO8859-1 ^a en_US.ISO8859-15 en_US.cp850. en_US.UTF-8, en_US.UTF-8@euro ^b
イギリス英語	en_GB.ISO8859-1 ^a en_GB.ISO8859-15 en_GB.UTF-8
ヨーロッパ	en_EU.UTF-8@euro ^c
フィンランド語	fi_FI.ISO8859-1 ^a fi_FI.ISO8859-15 fi_FI.UTF-8
フランス語	fr_FR.ISO8859-1 ^a fr_FR.ISO8859-15 fr_FR.UTF-8
ベルギー系フランス語	fr_BE.ISO8859-1 ^a fr_BE.ISO8859-15 fr_BE.UTF-8
カナダ系フランス語	fr_CA.ISO8859-1 ^a fr_CA.ISO8859-15 fr_CA.UTF-8
スイス系フランス語	fr_CH.ISO8859-1 ^a fr_CH.ISO8859-15 fr_CH.UTF-8
ドイツ語	de_DE.ISO8859-1 ^a de_DE.ISO8859-15 de_DE.UTF-8
スイス系ドイツ語	de_CH.ISO8859-1 ^a de_CH.ISO8859-15 de_CH.UTF-8
ギリシャ語	el_GR.ISO8859-7 el_GR.ISO8859-7@ucs4 el_GR.UTF-8
ヘブライ語	he_IL.ISO8859-8 he_IL.ISO8859-8@ucs4
ハンガリー語	hu_HU.ISO8859-2 hu_HU.ISO8859-2@ucs4

表 8-1: 言語とロケール (続き)

言語	ロケール名
アイスランド語	is_IS.ISO8859-1 ^a is_IS.ISO8859-15
イタリア語	it_IT.ISO8859-1 ^a it_IT.ISO8859-15 it_IT.UTF-8
日本語	ja_JP.eucJP ja_JP.SJIS ja_JP.SJIS@ucs4 ja_JP.deckanji ja_JP.deckanji@ucs4 ja_JP.sdeckanji ja_JP.UTF-8
韓国語	ko_KR.deckorean ko_KR.deckorean@ucs4 ko_KR.eucKR ko_KR.KSC5601 ko_KR.UTF-8
リトアニア語	lt_LT.ISO8859-4 lt_LT.ISO8859-4@ucs4
ノルウェー語	no_NO.ISO8859-1 ^a no_NO.ISO8859-15 no_NO.UTF-8
ポーランド語	pl_PL.ISO8859-2 pl_PL.ISO8859-2@ucs4
ポルトガル語	pt_PT.ISO8859-1 ^a pt_PT.ISO8859-15 pt_PT.UTF-8
ロシア語	ru_RU.ISO8859-5 ru_RU.ISO8859-5@ucs4
スロバキア語	sk_SK.ISO8859-2 sk_SK.ISO8859-2@ucs4
スロベニア語	sl_SI.ISO8859-2 sl_SI.ISO8859-2@ucs4
スペイン語	es_ES.ISO8859-1 ^a es_ES.ISO8859-15 es_ES.UTF-8
スウェーデン語	sv_SE.ISO8859-1 ^a sv_SE.ISO8859-15 sv_SE.UTF-8

表 8-1: 言語とロケール (続き)

言語	ロケール名
タイ語	th_TH.TACTIS
トルコ語	tr_TR.ISO8859-9 tr_TR.ISO8859-9@ucs4

^aこのロケールは、オプションのワールドワイド言語サポート・サブセットではなくベース・インストレーションに含まれます。

^ben_US.UTF-8@euro ロケールは、ローカル通貨記号をユーロ文字に定義し、国際通貨記号を EUR に定義します。

^cen_EU.UTF-8 ロケールは、ローカル通貨記号をユーロ文字に定義し、国際通貨記号を EUR に定義します。また、小数点をコンマ (,) と定義し、千の位の区切り文字をピリオド (.) と定義します。特に LC_MONETARY ロケール・カテゴリあるいは環境変数に定義する場合、このロケールは、英語を話す国だけでなく多くのヨーロッパの国々にとって便利です。

言語あるいは文字セットは必要に応じて切替えることができ、同一システム上で同時に複数のプロセスで別々の言語あるいはコードセットを使用することも可能です。

ISO8859-9 など、特定のコード化文字セットについての詳細はリファレンス・ページを参照してください。 UCS-4 および UTF-8 エンコーディングについては、Unicode(5) を参照してください。 PC コード・ページについては、code_page(5) を参照してください。

8.3 ロケールの作成

localedef ユーティリティによってプログラマは、独自のロケールを作成し、ソース・コードをコンパイルし、新しいロケールの固有名を生成できます。

ロケール作成の詳しい説明は、『国際化ソフトウェア・プログラミング・ガイド』を参照してください。

8.4 コードセット変換

オペレーティング・システムには、iconv ユーティリティおよび iconv_open(), iconv(), iconv_close() 関数が提供されています。これらの機能を使用することによりテキストを別のコードセットに変換することができ、これにより、プログラマが国際化アプリケーションを作成するのに支援します。オペレーティング・システムには、これらのインタフェースで利用できる多数のコードセット・コンバータが用意されています。

新しい `en_US.UTF-8` X ロケール・データベースには、オペレーティング・システムで使用する各種のフォントを含むフォント定義が入っています。このため、`en_US.UTF-8` ロケール下で実行されているアプリケーションは、ワールドワイド言語サポート (WLS) でインストールされたすべてのフォントの文字を表示できます。アジア言語系ロケール下で実行されているアプリケーションは、`ISO8859-2`、`-4`、`-5`、`-7`、`-8`、`-9`、および `TACTIS` 以外の、WLS でインストールされたすべてのフォントを表示できます。

同じ言語の異なるコードセット間の変換に加えて、これらのコンバータは、`UCS-2`、`UCS-4`、および `UTF-8` などの、異なる Unicode 間の変換もサポートします。また、一般的に使用される PC コード・ページ形式を処理するコードセット・コンバータも提供されています。

コードセット変換機能は、プリント・サブシステムや `man` コマンドなどのユーティリティでも使用されています。また、コードセット変換は、異なるコードセットを使用するシステム間でメールの交換を行うためにメール・ユーティリティにも実装されているとともに、テキスト入力、描画およびクライアント間通信のために、`X Window System Toolkit` にも実装されています。コードセット変換の詳細については、`iconv_intro(5)` リファレンス・ページを参照してください。Unicode および PC コード・ページ用のコンバータの詳細については、`Unicode(5)` および `code_page(5)` リファレンス・ページをそれぞれ参照してください。

8.5 Unicode および dense コード・ロケール

システム上にワールドワイド言語サポートをインストールすると、2 種類のロケール (Unicode ロケールと dense コード・ロケール) でローカリゼーションがサポートされます。

Unicode ロケールは、Unicode 標準と `ISO/IEC 10646` 標準に適合しており、ワイド文字のコード化に `UTF-32` を使用します。`UTF-32` のワイド文字エンコーディングでは、`wchar_t` の値は、ロケールに関係なく同じ文字を表します。Unicode 標準は広く普及しているため、どのプラットフォームでも実装は一貫しています。

dense コード・ロケールでは、テーブル・サイズを最小にするため、ワイド文字のコード化に dense コードが使用されます (つまり、コードポイントは、空のポジションなしで連続して割り当てられます)。

文字の内部表現および外部表現の両方に ISO 10646 (Unicode) を使用する UTF-8 ロケールに加え、dense コード・ロケールおよび Unicode ロケールは、機能的には同じ多くのロケールを提供します。

dense コード・ロケールのロケール名は、UTF-8 以外の文字列で終わります (たとえば ISO8859-1, eucJP, GB18030 など)。非 UTF-8 の Unicode ロケールのロケール名は、@ucs4 の文字列で終わります。dense コード・ロケールと Unicode ロケールのペアの例としては、pl_PL.ISO8859-2 と pl_PL.ISO8859-2@ucs4 があります。

一般に、dense コード・ロケールと Unicode ロケールには同じ charmap とロケール・ソースが使われます。ただし、ロケール・ソースの LC_COLLATE セクションで定義されていない文字は、2 つのタイプのロケールでソート結果が異なる場合があります。

Latin-1 ロケール (ISO8859-1) では、Latin-1 の文字が Unicode の最初の 256 文字と同じであるため、dense コード・ロケールと Unicode ロケールは全く同じです。オペレーティング・システムは、Unicode 標準で定義されている UTF-8、UTF-16、および UTF-32 の UTF (UCS transformation format) もサポートします。詳細はUnicode(5)を参照してください。

Unicode と dense コード・ロケールの切り替えは、root ユーザとして `i18nconfig` を実行してシステム全体のデフォルト設定を変更するか、あるいはシンボリック・リンク `/usr/i18n/lib/nls/dloc` を `./ucslloc` から `./loc` へ手動で変更することにより可能です。

8.6 Unicode サポート

Tru64 UNIX は、UCS-4 および UTF-8 ベース・ロケールで Unicode 標準 Version 3.1 および ISO 10646 標準をサポートします。UCS-4 (UTF-32)、UCS-2 (UTF-16)、および UTF-8 フォーマット間のコードセット変換機能が、すべてのサポート・コードセットに対して提供されます。Unicode といくつかのシングルバイト PC コード・ページ間の変換サポート、および、それらの PC コード・ページから ISO Latin コードセットへの変換機能が提供されます。Unicode ロケールについての詳細はUnicode(5)を参照してください。

8.7 国際化ソフトウェア構成ユーティリティ

「国際化ソフトウェアの構成」ユーティリティを使うと、システム管理者は、各国語サポート・サブセット、アジア言語のターミナル・ドライバ、イ

インストール済みフォント・ファイル、ローカル言語設定と入力メソッド、ユーザ・アカウント、および日本語入力メソッド (Wnn) を管理できます。これらの WLS オプションを設定することで、国際化アプリケーションの作成や使用に必要なオペレーティング・システム環境の設定を行うことができます。「国際化ソフトウェアの構成」ユーティリティを使用して、システム管理者やユーザはキーボード・マッピングを表示することもできます。

「国際化ソフトウェアの構成」ユーティリティは、SysMan Menu の「ソフトウェア」オプションから利用できるメニュー形式のユーティリティです。

- 各国語用のインストール済みのサブセットの表示および削除
- アジア系言語用のターミナル・ドライバのサポート、タイ語のサポート、擬似ターミナル・ドライバ、UTX (UNIX Terminal Extension) デバイスの数などのサポート・オプションの設定と変更後のカーネルの再構築
- インストール済みフォントの表示および削除
- インストール済みキーボード・マップ・ファイルの表示
- インストール済みロケールの表示、システムのデフォルト・ロケールの変更、dense コード・ロケールと Unicode ロケールの切り替え、ロケールの入力メソッドの選択
- WLS サポートのためのユーザ・アカウント、root アカウント、およびシステム・アカウントの構成
- Wnn の構成

8.8 ユーロ文字のサポート

Tru64 UNIX はユーロ通貨記号をサポートします。UTF-8 あるいは Latin-9 (ISO 8859-15) コードセットをサポートするロケールでユーロ文字をサポートする一方、ロケール名の最後に @euro が付くロケールではローカル通貨記号がユーロ文字になるように定義しています。

en_EU.UTF-8@euro ロケールは、ユーロ記号をサポートする英語ロケールです。ユーロ文字のプリンタ・サポートは汎用 PostScript プリント・フィルタ `wwpsmf` で提供されます。

ユーロ文字のキーボード入力は、キーマップで定義されているキー・シーケンスおよび Compose キーの使用によってサポートされます。詳細は `euro(5)` および `wwpsmf(8)` リファレンス・ページを参照してください。

8.9 dxim 入力サーバ

マルチリンガル入力サーバ dxim は、韓国語および中国語 (繁体字および簡体字) の入力メソッドの使用および管理のための手段を提供します。

dxim 入力サーバのメニューは、入力メソッド・クラスおよびメソッドのカスタマイズ、および、入力メソッド・ウィンドウのカスタマイズの 2 つの機能部分に分かれています。

- 入力メソッド・クラスおよびメソッドのカスタマイズでは、以下のような操作を実行できます。
 - クライアント・アプリケーションのロケールに適した入力メソッド・クラスの選択。中国語対応のアプリケーションに対しては、繁体字、簡体字、Phrase のクラスから 1 つあるいは複数をユーザが選択し起動します。
 - クラス内の 1 つあるいは複数の入力メソッドの選択および起動。Phrase 入力メソッドを除き、dxim が制御する繁体字および簡体字中国語クラスは、dxhanziim および dxhanyuim として一連の同じ入力メソッドをサポートします。Phrase 入力メソッドは dxim が制御する別のクラスで、オペレーティング・システムの Phrase ユーティリティが使うのものとは別のデータベースを使用します。
 - 省略時の入力メソッド・クラスの設定
 - クラスに対する省略時の入力メソッドの設定
 - 簡体字中国語 5-Shape および Intelligent ABC 入力メソッド・クラスのカスタマイズ
 - エラー・ベルのボリュームのカスタマイズおよび入力メソッド起動キーの設定
- 入力メソッド・ウィンドウのカスタマイズでは、以下のような操作を実行できます。
 - root 入力ウィンドウのフォント・サイズの変更
 - root 入力ウィンドウの前景色および背景色の設定
 - root 入力ウィンドウの行間隔の設定

dxim 入力サーバは、異なるロケールで動作する複数のクライアントをサポートできます。クライアント・アプリケーションが dxim に接続すると、

入力サーバがクライアントのロケールを確認し、そのまま使用できる場合は省略時の入力メソッドを使用します。クライアントのロケールではサーバの省略時の入力メソッドを使用できない場合、`dxim` は使用可能な入力メソッドを探します。この場合、最初に見つかった利用可能な入力メソッドが使用されます。

`dxim` 入力サーバについての詳細は、`dxim(1X)` リファレンス・ページおよび `dxim` のオンライン・ヘルプを参照してください。

8.10 国際化 Curses ライブラリ

オペレーティング・システムは、X/Open Curses, Issue 4 に準拠する国際化 Curses ライブラリを提供します。このライブラリは、シングルバイトおよびマルチバイト文字を処理するための関数を提供します。マルチバイト文字には、ワイド文字 (`wchar_t`) あるいは複合文字 (`cchar_t`) のどちらかのフォーマットが含まれます。複合文字フォーマットは、複数のワイド文字で構成されるシングル・ロジカル文字のために提供されます。複合文字のいくつかのコンポーネントは、ノンスペース文字です。

すべての Curses インタフェースの使用方法および機能については、`curses(3)`を参照してください。国際化 Cursesルーチンによって提供される機能の向上と、以前の Curses ルーチンとの関係については、『国際化ソフトウェア・プログラミング・ガイド』を参照してください。

8.11 その他の国際化機能

Tru64 UNIX は、以下の国際化ユーティリティおよび機能をサポートします。

- ベース `tty` 端末ドライバ・サブシステム

このサブシステムは、日本語、中国語、韓国語、タイ語のデータ処理のための追加の BSD ライン・ディシプリンと STREAMS 端末ドライバ・モジュールを含みます。たとえば、拡張端末サブシステムは、これらの言語に関して次のような機能をサポートします。

- 日本語かな漢字変換入力メソッド
- cooked モードでの文字ベースのライン処理
- 入力行の履歴と編集 (BSD ライン・ディシプリンのみ)
- ユーザ定義文字のソフトウェア・オンデマンド・ローディング
- 端末コードとアプリケーション・コードの変換

- `asort` ユーティリティ

`sort` コマンドを拡張したこのコマンドは、日本語や中国語などの表意文字をさまざまな照合順序でソートします。`asort` ユーティリティについての詳細は、`asort(1)` を参照してください。

- マルチリンガル Emacs エディタ (MULE)

Mule は、GNU Emacs を拡張してマルチリンガル対応したものです。このエディタは、GNU Emacs の機能に加えて、さまざまな言語の文字を表示、入力、編集する機能を提供します。詳細については `mule(1)` を参照してください。

- 日本語/中国語/韓国語のユーザ定義文字

`cedit` および `cgen` ユーティリティにより、ビットマップ・フォントも含め、ユーザが文字フォントとその属性を作成および定義することができます。X クライアントが、ユーザ定義文字のビットマップ・フォントを入手するために X サーバあるいはフォント・サーバを通して UDC データベースを使用できるように、フォント・レンダリング機能が提供されています。

ユーザ定義文字については、『国際化ソフトウェア・プログラミング・ガイド』、`cedit(1)`、および `cgen(1)` を参照してください。

- 種々の言語のプレーン・テキストおよび PostScript ファイルの印刷

Tru64 UNIX は、PostScript プリンタで高品質の印刷を行うためのアウトライン・フォントを提供しています。ローカル言語対応のプリンタをサポートするためのプリント・フィルタに加え、汎用の国際化プリント・フィルタでさまざまなプリンタをサポートとしています。この種のフィルタの 1 つとして、`wwpsof` は、必要なフォントを持たない PostScript プリンタで、ローカル言語ファイルの印刷をサポートします。

国際化プリント機能の詳細については、`il8n_printing(5)`、`pcfof(8)`、および `wwpsof(8)` リファレンス・ページを参照してください。

- メールと 8-bit 文字のサポート

省略時の設定では、Tru64 UNIX は、`mailx`、`dtmail`、`MH` および `comsat` で 8 ビット文字エンコーディングをサポートします。これらのメール・ユーティリティについての詳細は、`mailx(1)`、`dtmail(1)`、`mh(1)`、および `comsat(8)` を参照してください。

- `file` コマンド

`file` コマンドは、どのロケール設定でも UCS-2 および UCS-4 エンコーディングを認識するように拡張されています。他のエンコーディング・フォーマットに関しては、現在のロケール設定に対して有効な場合はファイル・データ・エンコーディングを認識します。このコマンドには `jfile` エイリアスも用意されています。 `jfile` エイリアスは、DEC Kanji、日本語 EUC、シフト JIS、および 7-bit JIS エンコーディングを認識します。

- グラフィカル・アプリケーションの国際化

Motif Version 1.2.3 の国際化機能は、ロケールのサポートのために、X11R6 および C ライブラリの多くの国際化機能を利用しています。Motif Version 1.2.3は、非 ISO Latin-1 キーストロークの入力を可能にする入力メソッドをサポートし、大幅に改良された `XmText` ウィジェットを提供することで、マルチバイトおよびワイド文字フォーマット、on-the-spot 入力スタイルをサポートしています。

Motif は、国際化 X ライブラリ関数および C ライブラリ関数により、マルチバイト文字およびワイド文字エンコーディングをサポートします。また、コンパウンド・ストリング関数は、ローカライズされた文字の作成を可能にする X11R6 `XFontSet` コンポーネントを含みます。

ユーザ・インタフェース言語 (UIL) は、UIL コンパイラの新しいコンパイル時スイッチである `-s` によって、ロケール対応の UID ファイルの作成をサポートしています。

代替入力メソッドは、`VendorShell` ウィジェットのリソースによって指定することができます。親ウィジェットとして `Shell` クラス・ウィジェットを持つウィジェットはこのリソースを使用することができ、入力のための特定のメソッドに対してそれらを登録することができます。

ウィンドウ環境

この章では、Common Desktop Environment (CDE)、および Tru64 UNIX ウィンドウ環境の X11R6.3 と Motif の構成要素について説明します。次のトピックについて説明します。

- Common Desktop Environment (9.1 節)
- X Window System (9.2 節)
- Motif Version 1.0 のコンポーネント (9.3 節)

9.1 Common Desktop Environment

Common Desktop Environment (CDE) は、X コンソーシアムの X ウィンドウ・システムおよび OSF の Motif ユーザ・インタフェースなどの業界標準をベースに共同開発されたグラフィカル・ユーザ・インタフェースです。CDE は、マルチベンダ・プラットフォーム上で、共通な API とともに一貫したルック・アンド・フィールを提供します。

CDE は、ユーザによるカスタマイズが可能なビジュアルなデスクトップを提供します。CDE インタフェースを使用すると、マウスとキーボードを使用してアプリケーションの操作を行うことができます。CDE の画面上の下部にフロント・パネルが表示され、アプリケーション、プリンタ、頻繁に使用するオブジェクト、オンライン・ヘルプなどにアクセスできます。

ユーザに対するサービスに加えて、CDE には、アプリケーションを CDE に統合するために必要なものがすべて用意されています。CDE は標準に準拠しているため、統合されたアプリケーションは、この標準に準拠する他のプラットフォームに容易にポータリング可能です。たとえば、ヘルプ・ファイルとそれにアクセスする方法は、標準に準拠するすべてのプラットフォームに対して適用できます。詳細については、『*Common Desktop Environment: プログラマ概要*』を参照してください。

CDE のフロント・パネルには、アプリケーションの起動、デスクトップ・セッションでの管理作業、ワークスペースの変更などを行うためのツール

のアイコンが表示されます。ワークスペースとは、フロント・パネルが表示されるスクリーンのことです。フロント・パネルは、別のワークスペースへ移動しても表示されます。フロント・パネルには次のツールのアイコンが配置されています。

- 時計
- カレンダー
- ファイル・マネージャ
- テキスト・エディタ
- メーラー
- ロック
- ワークスペース・スイッチ
- ビジー・ライト
- プリンタ
- セッション終了
- SysMan Menu
- スタイル・マネージャ
- アプリケーション・マネージャ
- ヘルプ・マネージャ
- ゴミ箱

各ツールの使用方法については、『*CDE ガイドブック*』および『*Common Desktop Environment: ユーザーズ・ガイド*』を参照してください。

CDE には CDE Window List が組み込まれています。CDE Window List は、マウス・ポインタが root ウィンドウにあるときにマウスの真中のボタンをクリックすることにより起動できます。このツールは、デスクトップの各ワークスペース上にあるアプリケーション・ウィンドウを検索し、起動中のアプリケーションの名前、クラス、ワークスペース名を表示します。詳細は、CDE Window List のオンライン・ヘルプを参照してください。

CDE Setup 機能は、GUI により CDE の設定とカスタマイズを容易にします。また、フロント・パネル制御の作成も容易になります。フロント・パネル制御は、フロント・パネル・タイプに組み入れることができます。これら

のタイプは、別のユーザのためにカスタマイズすることができます。さらに、ユーザは、CDE ウィンドウ、アイコン、バインディングに対してパラメータを設定し、`dtterm`、`dtmail`、`dtfile` アプリケーションのオプションを設定し、フロント・パネル端末エミュレータを決めることができます。

X サーバ、ログイン画面、システム・サービスの構成など、システム管理者用の機能も拡張されています。CDE Setup についての詳細は、`dtsetup(8)` リファレンス・ページ、および CDE Setup のオンライン・ヘルプを参照してください。

9.2 X ウィンドウ・システム

X Window System Version 11, Release 6.3 のソフトウェアは、次のコンポーネントで構成されています。

- X クライアント・ライブラリ
- X サーバ
- ディスプレイ・マネージャ
- X プロトコル拡張機能
- フォント・サーバ
- X クライアント
- X プリント・サーバ (Xp)

9.2.1 X クライアント・ライブラリ

Tru64 UNIX は、X11R6.3 の X クライアント・ライブラリを完全にサポートしています。

- Athena ウィジェット・セット (`libXaw`)
ユーザ・インタフェース・コンポーネント (スクロール・バー、ラベル、ボタン) の高レベル・ライブラリ
- X イントリンシックス・ライブラリ (`Xt`)
`xlib` にコールを行う中レベル・ルーチン
- X ライブラリ (`xlib`)
X サーバとのインタフェースを行う低レベル・ルーチン

X クライアント・ライブラリについての詳細は、X Window のドキュメントを参照してください。

9.2.2 X サーバ

シェアード・ライブラリを多用することで、Tru64 UNIX はすべてのグラフィック・オプションに対して単一の X11R6.3 X サーバ・イメージをサポートしています。Tru64 UNIX のX サーバは初期化時に動的に構成を行い、特定のシステム構成で要求されるサーバ・コンポーネントだけをロードします。システム管理者の作業を必要とすることはほとんどありません。

9.2.3 マルチヘッド・グラフィック・サポート

適切なオプション・カードがインストールされ、サポート機能がカーネルに組み込まれている場合、Tru64 UNIX におけるマルチヘッド・グラフィック・サポートは透過的です。X11R6.4 の Xinerama として知られる Panoramix 拡張により、マルチヘッド・システムをより適切に利用できます。

9.2.4 X サーバ拡張機能

Tru64 UNIX は次の X サーバ拡張機能をサポートしています (省略時の設定では、メモリを節約するために、X サーバはクライアントから特定の拡張機能への要求が出されるまで、ほとんどのサーバ拡張機能のロードを行いません)。

- X11R6.3 に対するキーボード拡張機能

X11R6.3 に対するキーボード拡張 (XKB) により、以下のような機能を提供して X Window System におけるキーボードの制御およびカスタマイズ機能を強化しています。

- キーボード・レイアウトに関する ISO 9996 標準のサポート
- コア X キーボードの処理との互換性 (クライアントの修正は不要)
- キーボードの LED 処理 や CapsLock および NumLock などのロック処理のための標準の方法
- キーボード・ジオメトリのサポート

また、障害を持つ人々のための X11R5 AccessX サーバ拡張機能が XKB サーバ拡張機能に組み込まれています。これらの機能としては、オートリピートに関する制御とともに StickyKeys, SlowKeys, BounceKeys, MouseKeys および ToggleKeys が含まれます。

- **XKME (X Keyboard Management Extension)**

国際化 X クライアントをより適切にサポートするための内部拡張機能です。XKB 拡張機能によって、XKME 機能は古い機能となっていますが、下位互換性のために提供されています。

- **MIT-SHM (MIT 共有メモリ)**

イメージを多用するローカル・アプリケーションの性能を向上させます。

- **MIT-SUNDRY-nonstandard**

X コンソーシアムの拡張機能です。現在は X サーバに対するバグ互換性モードの制御に使用されます。

- **マルチバッファリング**

複数バッファに書き込むことで、スムーズなアニメーションをサポートします。

- **SHAPE**

円形、楕円形あるいは特殊な形のウィンドウなど、四角形以外のウィンドウをサポートします。

- **SMT (共有メモリ・トランスポート)**

ローカル・クライアントの X トランスポートとして共有メモリを使用することを可能にし、性能を向上させます。トランスポートは `local:0.0` で指定します。

- **XIE (X Imaging Extension, Version 3.0)**

イメージの拡張制御およびデバイスに依存しないイメージ表示を提供します。

Tru64 UNIX では、バージョン 3 (`/usr/lib/Xie.a` および `/usr/shlib/libXie.so`) と業界標準のバージョン 5 (`/usr/lib/libXIE.a` および `/usr/shlib/libXIE.so`) を提供します。

- **X-Input**

他社の入力デバイスに対応する、ユーザ独自のドライバの作成を可能にします。作成したドライバは、X サーバ構成ファイル (`/usr/var/X11/Xserver.conf`) のエントリに追加することによって X サーバに動的ロードできます。新しい入力デバイスは、X サーバが次にリセットされた時点で認識されるようになります。

従来の静的リンク X サーバは、新しい拡張デバイスを追加するたびに X サーバの再構築が必要です。Tru64 UNIXのロード可能 X サーバの実装では、X サーバ初期化時にロードされる外部共用可能デバイスとして新しい入力デバイスをシステム管理者が追加できるようにすることによって、この制限を回避しています。

ドライバのサンプル・コードは `/usr/examples` ディレクトリに用意されています。

- X スクリーン・サーバ

一定の期間スクリーンが使用されていない場合、または再度スクリーンが使用されたとき、クライアントに通知します。この拡張機能はスクリーン・サーバ・アプリケーションの開発に役に立ちます。

- XSync

XSync 関数を XFlush、XEventsQueued および XPending 関数とともに使用すると、X サーバの内部で X クライアント間の同期を取ることが可能になります。これにより、ネットワークにより引き起こされるエラーを除去することができ、また、異なるオペレーティング・システムが実行されているホスト間で X クライアントの同期を取ることができます。この拡張機能は、特にオーディオ、ビデオおよびグラフィックスなどの同期が要求されるマルチメディア・アプリケーション、および要求を内部 X サーバ・タイマと同期させるようなアニメーション・アプリケーションに効果的です。

- XTest

テストのために、アプリケーションが X イベントをシミュレートすることが可能になります。

- XTrap

X クライアントのテストのための X イベントの記録および再生をサポートします。

- XV (X Video)

ライブ・ビデオ PIP オプションなどの、TX グラフィック・デバイスに関するビデオ・オプションをクライアントが制御することが可能になります。

- X プリント拡張

この機能により、X アプリケーションがプリント・デバイスへ直接出力することが可能になります。

- X プリント・サーバ (Xp)

これによって、X イメージングがプリンタやファックスのような装置を非表示にできます。X Print サービスの中心となるのは、X Print Server です。印刷オペレーションが必要なアプリケーションは X Print Server に接続し、利用可能なプリンタの一覧表示、プリンタの選択、印刷要求のキュー登録を実行できます。

- Panoramix

システムに複数のビデオ・モニタを構成し、それらを 1 つの大きなスクリーンとして機能させることが可能です。ウィンドウは複数のスクリーンに広がることも、1 つのスクリーンから別のスクリーンへ移動することもできます。この拡張機能は、同種のグラフィック環境でのみサポートされます。この環境は、共通のデバイス、深度、解像度などで構成されていなければなりません。

- アプリケーション・グループ

アプリケーション・グループ・ウィンドウがそのグループ内のウィンドウに対する特定の要求を取り出し、それらを適切に処理することができるよう、ウィンドウのグルーピングを可能にします。この機能はリモート実行拡張で使用され、X アプリケーションを Web ページに組み込むことを可能にしています。

- X セキュリティ

X Window System に対してエンハンスド・セキュリティ機能を提供します。セキュリティは、ユーザごと、あるいはリソースごとに指定することができます。

- リモート実行

Web ブラウザなどから、X アプリケーションをリモートで起動させることができます。

- 低帯域幅 X (lbx)

モデムなど低帯域幅転送で X をより効率的に動作させることができます。

個々の X クライアント・ライブラリについては、X Window のドキュメント、および、x(1X)、xdec(1X) リファレンス・ページを参照してください。

9.2.5 ディスプレイ・マネージャ

Tru64 UNIX は、標準の `xdm` 端末マネージャ・ソフトウェアをサポートしています。 `xdm` 端末マネージャは X サーバをローカルに起動し、ネットワーク透過のログイン・プロンプトを表示します。この機能により、`xdm` でサポートされているネットワーク上のすべてのシステムに対し、そのリモート・システムのグラフィック・コンソールで直接ログインするのと同様にログインできます。この機能により、X 端末を Tru64 UNIX 環境にスムーズに統合することができます。 `xdm` の詳細については、『システム管理ガイド』および `xdm(1X)` リファレンス・ページを参照してください。

キーマップ・フォーマット

Tru64 UNIX で提供する省略時のキーマップは、XKB 標準キーマップ・フォーマットです。これらのキーマップは、XKB 拡張をサポートおよび実行する X サーバで使用されます。XKB キーマップ・ファイルは、簡単にカスタマイズおよびコンパイルできるテキスト・ファイルです。

下位互換性を維持するため、`xmodmap` キーマップ・フォーマットもサポートされます。`xmodmap` キーマップ・フォーマットは事実上の業界標準で、シンボリック・キー名を使用して記述され、カスタマイズが容易です。

どちらのキーマップ・フォーマットも、モディファイア・キー (Compose, Alt, Shift など) を指定することができます。

`xmodmap` キーマップ・フォーマットではなく、XKB 規格キーマップ・フォーマットを使用することをお勧めします。

XDM-AUTHORIZATION-1

X クライアントが X サーバとの接続を確立する時に、キーと呼ばれる登録コードがクライアントからサーバへ渡されます。X サーバがこのキーを認識すると接続が認められます。`xdm` (X ディスプレイ・マネージャ) は、ユーザの X セッション起動時に 1 つまたは複数のキーをユーザのホーム・ディレクトリの `.Xauthority` ファイルに書き込みます。また、X ディスプレイ・マネージャ (`xdm`) は、X サーバが読み取り可能なファイルにもキーを書き込みます。

セキュリティの向上のため、Tru64 UNIX は MIT-MAGIC-COOKIE-1 キー・フォーマットと XDM-AUTHORIZATION-1 暗号化キー・フォーマットの両方をサポートしています。省略時のキーは XDM-AUTHORIZATION-1 です。

9.2.6 フォント・サーバ

Tru64 UNIX は、標準のスケラブル・フォント・サーバをサポートしています。これにより、ネットワーク上の各システムが、ネットワーク上のいずれかの Tru64 UNIX システムに常駐するフォントに対してアクセスする機能が提供されます。フォント・サーバは、フォントの格納場所を管理し、ネットワーク上の他の X サーバからローカルでは持っていないフォントを要求されたときにこれに応答します。フォントに対するネットワーク透過のアクセスを提供する他に、要求を出している X サーバにフォントを提供する前にフォントのスケリングを適切に行う機能を持っているため、フォント・サーバは、ローカル X サーバのフォント・スケールの計算負荷を軽減します。

フォントを X サーバに表示する前に、グリフをディスク上のフォーマットからビットマップに変換する必要があります。この変換は X サーバまたは X サーバにフォントを提供するフォント・サーバ内のフォント・レンダ・コードによって行われます。

Tru64 UNIX はロード可能フォント・レンダをサポートしているので、X11R6.3 標準に準拠した固有のフォント・セットのためのフォント・レンダを作成し、Tru64 UNIX にインストールすることができます。フォントおよびフォント・レンダのインストール後、必要なエントリが X サーバ構成ファイル (/usr/var/X11/Xserver.conf) あるいはフォント・サーバ構成ファイル (/usr/var/X11/fs/config) のどちらか一方または両方に置きます。新しいフォント・レンダは、次に X サーバまたはフォント・サーバ (フォント・レンダが構成されている方) がリセットされると認識されます。

9.2.7 X クライアント

Tru64 UNIX は、下記のような X11R6.3 で提供されるすべての X クライアントをサポートします。 appres, atobm, bdfpc, bitmap, bmta, chooser, editres, fs, fsinfo, fsfonts, fstobdf, iceauth, ico, imake, listres, lndir, maze, mkfontdir, oclock, optacon, puff, puzzle, restart, resize, showfont, showrgb, smproxy, twm, uil, viewres, x11perf, x11perfcomp, xauth, xbiff, xcalc, xcd, xclipboard, xclock, xcmsdb, xconsole, xcutsel, xdm, xdpr, xdpinfo, xedit, xemacs, xev, xeyes, xfd, xfindproxy, xfw, xfontsel, xfw, xhost, xkbells, xkbcomp, xkbprint, kbdfmap, , kbvleds, kbwatch, xkill, xload, xlogo, xlsatoms, xlsclients, xlsfonts, xmag, xman, xmbind, xmh,

xmkmf, xmodmap, xon, xpr, xprop, xrdb, xrefresh, xset, xsetroot, xsoundsentry, xstdcmap, xterm, xwd, xwininfo, および xwud。

各 X クライアントについての詳細は、それぞれのリファレンス・ページを参照してください。

9.3 Motif コンポーネント

Tru64 UNIX は、下記のような Motif V1.0 のすべてのコンポーネントをサポートしています。

- ウィジェット・ライブラリ (Xm)
- リソース・マネージャ (Mrm)
- ウィジェット・メタ言語 (wml)
- ユーザ・インタフェース言語 (UIL)
- Motif ウィンドウ・マネージャ (mwm)
- キー・バインド・ユーティリティ (xmbind)
- Motif デモンストレーション・プログラム (examples)

Open Software Foundation は Motif R1.2 で、ANSI C、国際化機能、ドラッグ・ドロップ機能および ティア・オフ・メニューのサポートを追加しました。これらのサポートにより、Motif R1.1.3 とのバイナリ互換性が維持できなくなるため、Tru64 UNIX ではバージョンング機能を通して Motif Version 1.1.3 ライブラリを提供し、Motif Version 1.1.3 にリンクされたアプリケーションが引続き実行できるようにしています。これらのライブラリはオプションのサブセットに含まれています。バージョンングについての詳細は、『プログラミング・ガイド』を参照してください。

Motif についての詳細は『*OSF/Motif Programmer's Guide*』および各リファレンス・ページを参照してください。Motif の国際化サポートについては第 8 章を参照してください。

9.3.1 拡張ウィジェット・セット

Xm ウィジェット・ライブラリに加えて、Tru64 UNIX では次のようなウィジェットを含む拡張ウィジェット・セット (DXm) をサポートしています。

- DXmColorMix

色の編集および選択をサポートします。

- DXmPrintWidget

グラフィック・プリント・オプションを提供します。

- DXmCSText

XmTextと同じようなユーザ・インタフェースで、コンパウンド・ストリングの編集をサポートします。

- DXmHelpWidget

ヘルプ・トピックを表示します。

- DXmSvn

データの一覧を通じて 構造化ナビゲーションをサポートします。

9.3.2 X クライアント

X11R6.3 で提供されるすべての X クライアントに加えて、Tru64 UNIX では、下記の X クライアントを提供しています。 accessx, dxconsole, dxdiff, dxkbledpabel, dxkeyboard, dxkeycaps, dxpresto, および dxterm。



UNIX と Windows 間の相互運用性

この章では、Windows システムと UNIX システム間のシームレスな対話を可能にする Tru64 UNIX アプリケーションについて説明します。次の各トピックについて説明します。

- Advanced Server for UNIX (10.1 節)
- アプリケーションからの ODBC 接続 および JDBC 接続を可能にする INTERSOLVE DataDirect ソフトウェア製品 (10.2 節)

10.1 Advanced Server for UNIX

Advanced Server for UNIX (ASU) ソフトウェアは、Tru64 UNIX と Windows 環境を統合する Tru64 UNIX レイヤード・アプリケーションです。ASU ソフトウェアは、Tru64 UNIX システム上で、Windows NT Server バージョン 4.0 のサービス、セキュリティ、および機能を実行します。ASU ソフトウェアが実行されている Tru64 UNIX システムは、他の Windows システムや Windows システムのユーザからは、Windows NT サーバと同じように認識されます。また、この Tru64 UNIX システムは Windows NT および Windows 2000 のドメインに参加できます。

ネイティブの Windows コマンドとユーティリティを使用して、ASU ソフトウェアを管理し、UNIX ベースのファイル・システム、ディレクトリ、およびプリンタを Windows ユーザと共有できます。Windows ユーザは、ソフトウェアを変更することなく、これらの共有リソースにアクセスできます。ASU による接続が完了すると、Tru64 UNIX のディレクトリとプリンタが、Windows ユーザのローカルのコンピュータ環境で透過的に認識されます。

ASU ソフトウェアは、Tru64 UNIX メディア・キットに含まれており、2 ユーザまでの接続が可能な無料ライセンスが提供されています。それ以上のユーザ数で使用したい場合の ASU ライセンスの購入について、および HP のライセンス条件およびポリシーについては、HP の各支店/営業所にお問い合わせください。

10.2 データ・アクセス (ODBC および JDBC)

Tru64 UNIX は、アプリケーションに対する ODBC および JDBC 接続を可能にする INTERSOLV DataDirect ソフトウェア製品ファミリを提供しています。このソフトウェアはアプリケーションの開発や導入のためのオプション・ソフトウェアであり、Tru64 UNIX オペレーティング・システムのライセンスの一部としてライセンスされています。

SequeLink ODBC Edition は、汎用 ODBC クライアント・コンポーネントです。DataDirect SequeLink ODBC は、ほとんどすべての種類のクライアント、ネットワーク、サーバ、またはデータベースに対する透過的な接続を実現します。

JDBC を使用して作成された Java アプリケーションは、別のプラットフォームのソースやデータベースにアクセスできます。SequeLink Java Edition は、汎用的な JDBC の標準ベースの実装です。柔軟性を備えており、マルチベンダ・クライアント、サーバ、および Web 環境から業界屈指のデータベースに至るまで、スケーラブルな接続性を提供します。Java 環境に合わせた最適化およびチューニングが施されているので、既存のシステムの機能性と性能が向上し、新しいテクノロジーを容易に統合できます。

A

インターネット標準

この付録は、Tru64 UNIX オペレーティング・システムが準拠しているインターネット RFC (Request for Comment) 標準および RFC 以外の標準の一覧です。

A.1 RFC 標準

表 A-1 は、Tru64 UNIX が準拠している RFC 標準の一覧です。インターネットに接続されているシステム上でオンラインでこのリストを参照している場合、RFC をクリックすると、Web ブラウザ内にその RFC のテキストを表示できます。

表 A-1: RFC 標準

RFC	プロトコル	名前
678		Standard File Formats
768	UDP	User Datagram Protocol (ユーザ・データグラム・プロトコル)
791		Internet Protocol (インターネット・プロトコル)
792	ICMP	Internet Control Message Protocol (インターネット制御メッセージ・プロトコル)
793	TCP	Transmission Control Protocol (伝送制御プロトコル)
819	SMTP	The Domain Naming Convention for Internet User Applications (インターネット・ユーザ・アプリケーションのドメイン命名規約)
821	SMTP	Simple Mail Transfer Protocol (シンプル・メール転送プロトコル)
822	MAIL	Standard for the Format of ARPA Internet Text Messages (ARPA インターネット・テキスト・メッセージ・フォーマットの標準)
826	ARP	Address Resolution Protocol (アドレス解決プロトコル)
854	TELNET	TELNET Protocol (TELNET プロトコル)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
855	TELNET	TELNET option specifications (TELNET オプション仕様)
856	TELNET	TELNET binary transmission (TELNET バイナリ伝送)
857	TELNET	TELNET echo option (TELNET エコー・オプション)
858	TELNET	TELNET Suppress Go Ahead option (TELNET Suppress Go Ahead オプション)
859	TELNET	TELNET status option (TELNET ステータス・オプション)
868	Time	Time Protocol (Time プロトコル)
893		Trailer Encapsulations
894	IP-E	Internet Protocol on Ethernet Networks
903	RARP	Reverse Address Resolution Protocol (逆アドレス解決プロトコル)
904	EGP	Exterior Gateway Protocol
919		Broadcasting Internet Datagrams (インターネット・データグラムのブロードキャスト)
920		Domain Requirements (ドメインの要件)
922		Broadcasting Internet Datagrams in the Presence of Subnets (サブネットが存在する場合のインターネット・データグラムのブロードキャスト)
950		Internet Standard Subnetting Procedure (インターネット標準のサブネット処理)
951	BOOTP	The Bootstrap Protocol
954		NICNAME/WHOIS (RFC 812 の置き換え)
959	FTP	File Transfer Protocol (ファイル転送プロトコル)
974		Mail Routing and the Domain System (メール・ルーティングとドメイン・システム)
1014	XDR	External Data Representation
1032		Domain Administrators Guide (ドメイン管理者ガイド)
1033		Domain Administrators Operations Guide (ドメイン管理者操作ガイド)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
1034		Domain Names (Concepts and Facilities) (ドメイン名 (概念と機能)) (RFC 882, 883, および 973 の置き換え)
1035	IP-IEEE	Domain Names (Implementation and Specification) (ドメイン名 (実装と仕様)) (RFC 882, 883, および 973 の置き換え)
1042	IP-IEEE	Internet Protocol on IEEE 802
1049		Content-Type Field for Internet Messages
1050	RPC	Sun Remote Procedure Calls (Sun リモート・プロシージャ・コール)
1055	SLIP	Serial Line Internet Protocol (シリアル・ライン・インターネット・プロトコル)
1057		RPC Protocol Specification Version 2 (RPC プロトコル仕様バージョン 2) (RFC 1050 の置き換え)
1058	RIP	Routing Information Protocol (ルーティング情報プロトコル)
1094	NFS	Network File System Protocol (ネットワーク・ファイル・システム・プロトコル)
1101		DNS Encoding of Network Names and Other Types (ネットワーク名と他のタイプの DNS 符号化) (RFC 1034 および 1035 の更新)
1112		Host Extensions for IP multicast
1116		Telnet Line Mode Option
1119	NTP	Network Time Protocol minus authentication
1122		Requirements for Internet Hosts – Communication Layers (インターネット・ホストの通信レイヤの要件)
1123		Requirements for Internet Hosts – Application and Support (インターネット・ホストのアプリケーションおよびサポートの要件)
1144	CSLIP	Compressing TCP/IP Headers for Low-Speed Serial Links
1155	SMI	Structure of Management Information
1156	MIB	Management Information Base
1157	SNMP	Simple Network Management Protocol (シンプル・ネットワーク管理プロトコル)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
1163	IP-FDDI	A Border Gateway Protocol (BGP) (境界ゲートウェイ・プロトコル (BGP))(RFC 1105 の置き換え)
1164		Application of the Border Gateway Protocol in the Internet, and OSPF/BGP Interaction (インターネット内の Border Gateway Protocol のアプリケーションと, OSPF/BGP の相互動作)
1183	IP-FDDI	New DNS RR Definitions (新しい DNS RR 定義) (RFC 1034 および 1035 の更新)
1188	IP-FDDI	Transmission of IP over FDDI (FDDI 上の IP の転送) (RFC 1103 の置き換え)
1191		Path MTU Discovery (ルータ仕様およびホスト仕様)
1212		Concise MIB definitions
1213	MIB-II	Management Information Base II (管理情報ベース II) (RFC 1158 の置き換え)
1231		IEEE 802.5 Token Ring MIB (セット・オペレーションのサポートなし)
1253		OSPF Version 2 Management Information Base (OSPF バージョン 2 管理情報ベース) (RFC 1252 の置き換え)
1256	ICMP	Router Discovery Messages
1267		A Border Gateway Protocol 3 (BGP-3) (境界ゲートウェイ・プロトコル 3 (BGP-3)) (RFC 1105 および 1163 の置き換え)
1269		Definitions of Managed Objects for the Border Gateway Protocol (Version 3), AS path pattern matching (パス・パターン・マッチングとしての, 境界ゲートウェイ・プロトコル (バージョン 3) の管理対象オブジェクトの定義)
1282		BSD rlogin
1285		FDDI Management Information Base (セット・オペレーションのサポートなし)
1288	FINGER	Finger Protocol (Finger プロトコル) (RFC 1196 , 1194 , および 742 の置き換え)
1305	NTP	Network Time Protocol Version 3.0 (ネットワーク・タイム・プロトコル・バージョン 3.0)
1321	MD5	The MD5 Message Digest Algorithm

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
1323	TCP-HIPER	TCP Extensions for High Performance (性能を高めるための TCP 拡張) (Window scale オプション, time stamp オプション, および PAWS)
1332	IPCP	The PPP Internet Protocol Control Protocol (RFC 1172 の置き換え)
1334	PAP/CHAP	PPP Authentication Protocols (PPP 認証プロトコル)
1350	TFTP	Trivial File Transfer Protocol (RFC 783 の置き換え)
1388	RIP	RIP Version 2 Carrying Additional Information (追加情報を運ぶ RIP バージョン 2)
1398		Definitions of Managed Objects for the Ethernet-like Interface Types (Ethernet ライクのインタフェース・タイプの管理対象オブジェクトの定義) (RFC 1284 の置き換え)
1403		BGP OSPF Interaction (BGP OSPF の相互動作) (RFC 1364 の置き換え)
1483		Multiprotocol Encapsulation over ATM Adaptation Layer 5 (ATM Adaptation レイヤ 5 上のマルチプロトコルのカプセル化) (ルーティングされたプロトコルのカプセル化のみ)
1514		Host Resources MIB (ホスト・リソース MIB) (セット・オペレーションのサポートなし)
1518	CIDR	An architecture for IP Address Allocation with CIDR (CIDR を利用した IP アドレス割り当てのためのアーキテクチャ)
1521		付録 A またはこの RFC で述べられている MIME のサポート
1533	DHCP	DHCP Options and BOOTP Vendor Extensions (DHCP オプションおよび BOOTP ベンダ拡張) (RFC 1497, 1395, 1084, および 1048 の置き換え)
1534		DHCP と BOOTP の相互運用
1535		A Security Problem and Proposed Correction With Widely Deployed DNS Software (広く使用されている DNS ソフトウェアでのセキュリティ上の問題と訂正案)
1536		Common DNS Implementation Errors and Suggested Fixes (DNS 実装に共通的に存在する誤りと修正案)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
1542		Clarifications and Extensions for the Bootstrap Protocol (Bootstrap プロトコルの明確化と拡張) (RFC 1532 の置き換え, および RFC 951 の更新)
1547	IS-PPP	Requirements for an Internet Standard Point-to-Point Protocol
1571		Telnet Environment Option Interoperability Issues (Telnet 環境オプションの相互運用性の問題) (RFC 1408 の更新)
1572		Telnet Environment Option
1577		Classical IP over ATM
1583	OSPF	OSPF V2 (RFC 1247 の置き換え)
1589		A Kernel Model for Precision Time-keeping (システム・クロックを外部のプレジジョン・タイミング・ソースに合わせる機能のサポートなし)
1591		Domain Name System Structure and Delegation (ドメイン・ネーム・システムの構造と権限の委任)
1597		Address Allocation for Private Internets (プライベート・インターネットのアドレス割り当て)
1626		Default MTU for IP over ATM
1627		Network 10 Considered Harmful (ネットワーク 10 の有害性)
1633		Integrated Services
1637		DNS NSAP Resource Records (DNS NSAP リソース・レコード) (RFC 1348 の置き換え)
1652	SMTP	Service Extension for 8-bit MIME transport
1654	PPP	A Border Gateway Protocol 4 (BGP-4) (境界ゲートウェイ・プロトコル 4 (BGP-4))
1661	PPP	The Point-to-Point Protocol (PPP) (ポイント・ツー・ポイント・プロトコル (PPP)) (RFC 1548 の置き換え) (非同期 IP のみ)
1700		Assigned Numbers (割り当て済み番号) (RFC 1340 の置き換え)
1713		Tools for DNS debugging (DNS デバッグ用ツール)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
1748		Token Ring interface information (トークン・リング・インタフェース情報) (RFC 1743 の置き換え)
1755		Signaling for IP of ATM
1794		DNS Support for Load Balancing (DNS の負荷バランスのサポート)
1813	NFS	Network File System Version 3 Protocol
1869	SMTP	Service Extensions
1870	SMTP	Service Extensions for Message Size Declaration (メッセージ・サイズ宣言のためのサービス拡張) (RFC 1653 の置き換え)
1876	DNS	A Means for Expressing Location Information in the Domain Name System (ドメイン・ネームシステムでの位置情報の表現方法) (RFC 1034 および 1035 の更新)
1884	IP V6	IP Version 6 Addressing Architecture (IP バージョン 6 のアドレッシング・アーキテクチャ)
1886		DNS Extensions to Support IP Version 6 (IP バージョン 6 をサポートする DNS 拡張)
1891	SMTP	Service Extensions for Delivery Status Notification
1892		Multipart/Report Content Type for the Reporting of Mail System Administrative Messages (メール・システム管理メッセージの報告用の Multipart/Report コンテンツ・タイプ)
1893		Enhanced Mail System Status Code
1894		Extensible Message Format for Delivery Status Notifications
1912		Common DNS Operational and Configuration Errors (DNS 共通の操作および構成のエラー) (RFC 1537 の置き換え)
1939	POP3	Post Office Protocol Version 3 (Post Office プロトコル・バージョン 3) (RFC 1725 の置き換え)
1948		Defending Against Sequence Number Attacks
1953	IFMP	Ipsilon Flow Management Protocol Specification for IPv4
1954		Transmission of Flow Labeled IPv4 on ATM Data Links

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
1981		Path MTU Discovery for IPv6 (IPv6 用のパス MTU 検出)
1985	SMTP	Service Extension for Remote Queue Starting
1995		Incremental Zone Transfer in DNS (DNS での増分ゾーン転送) (RFC 1035 の更新)
1996		A Mechanism for Prompt Notification of Zone Changes (DNS NOTIFY) (ゾーン変更の通知を促進するためのメカニズム (DNS NOTIFY)) (RFC 1035 の更新)
2001		TCP Slow Start , Congestion Avoidance , Fast Retransmit , Fast Recovery Algorithms
2010		Operational Criteria for Root Name Servers (ルート・ネーム・サーバの動作基準)
2018		TCP Selective Acknowledgement Options (TCP 選択肯定応答オプション)
2052		A DNS RR for specifying the location of services (DNS SRV) (サービスの位置を指定する DNS RR (DNS SRV)) (RFC 1035 および 1183 の更新)
2060	IMAP4	Internet Message Protocol , Version 4 Rev. 1 (インターネット・メッセージ・プロトコル・バージョン 4 リビジョン 1) (RFC 1730 の置き換え)
2080	RIPng	RIPng for IPv6 (IPv6 用 RIPng)
2131	DHCP	Dynamic Host Configuration Protocol (動的ホスト構成プロトコル) (RFC 1541 の置き換え)
2132	DHCP	DHCP options and BOOTP vendor extensions (DHCP オプションと BOOTP ベンダ拡張) (RFC 1533 の置き換え)
2136		Dynamic Updates in the Domain Name System (DNS UPDATE) (ドメイン・ネーム・システムでの動的更新 (DNS UPDATE)) (RFC 1035 の更新)
2137		Secure Domain Name System Dynamic Update (セキュア・ドメイン・ネーム・システムの動的更新) (RFC 1035 の更新)
2181		Clarifications to the DNS Specification (DNS 仕様の明確化) (RFC 1034 , 1035 , および 1123 の更新)
2182		Selection and Operation of Secondary DNS Servers (セカンダリ DNS サーバの選択と操作)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
2205	RSVP	Resource Reservation Protocol for FDDI and Ethernet
2210		Use of RSVP with IETF Integrated Services (IETF 統合サービスによる RSVP の使用)
2211		Controlled Load Services
2308		Negative Caching of DNS Queries (DNS NCACHE) (DNS 照会のネガティブ・キャッシュ (DNS NCACHE)) (RFC 1034 および 1035 の更新)
2373		IPv6 Addressing Architecture (IPv6 アドレッシング・アーキテクチャ) (anycast アドレッシングは、現在インプリメントされていない)
2374		An IPv6 Aggregatable Global Unicast Address Format (IPv6 集約可能グローバル・ユニキャスト・アドレス・フォーマット) (RFC 2073 の置き換え)
2401		Security Architecture for the Internet Protocol (RFC 1825 の置き換え)
2402		IP Authentication Header (RFC 1826 の置き換え)
2403		The Use of HMAC-MD5-96 within ESP and AH
2404		The Use of HMAC-SHA-1-96 within ESP and AH
2405		The ESP DES-CBC Cipher Algorithm with Explicit IV
2406		IP Encapsulating Security Payload (ESP)
2407		The Internet IP Security Domain of Interpretation for ISAKMP
2408		Internet Security Association and Key Management Protocol (ISAKMP)
2409		The Internet Key Exchange (IKE)
2410		The NULL Encryption Algorithm and Its Use With IPsec
2411		IP Security Document Roadmap
2412		The OAKLEY Key Determination Protocol
2460	IPv6	Internet Protocol, Version 6 (IPv6) Specification (インターネット・プロトコル・バージョン 6 (IPv6) 仕様) (RFC 1883 の置き換え)
2461		Neighbor Discovery for IPv6 (IPv6 用近隣検出) (RFC 1970 の置き換え)

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
2462		IPv6 Stateless Address Autoconfiguration (IPv6 ステートレス・アドレスの自動構成) (RFC 1971 の置き換え)
2463	ICMPv6	Internet Control Message Protocol (ICMPv6) for IPv6 (IPv6 用インターネット制御メッセージ・プロトコル (ICMPv6)) (RFC 1885 の置き換え)
2464		Transmission of IPv6 Packets over Ethernet Networks (Ethernet ネットワーク上の IPv6 パケットの転送) (RFC 1972 の置き換え)
2467		Transmission of IPv6 Packets over FDDI Networks (FDDI ネットワーク上の IPv6 パケットの転送) (RFC 2019 の置き換え)
2471		IPv6 Testing Address Allocation (IPv6 テスト用アドレスの割り当て) (RFC 1897 の置き換え)
2472		IPv6 over PPP (PPP 上の IPv6) (RFC 2023 の置き換え)
2535		Domain Name System Security Extensions (ドメイン・ネーム・システムのセキュリティ拡張) (RFC 2065 の置き換え, RFC 1035, 2181, および 2534 の更新)
2536		DSA KEYS and SIGs in the Domain Name System (DNS) (ドメイン・ネーム・システム (DNS) での DSA KEY と SIG)
2537		RSA/MD5 KEYS and SIGs in the Domain Name System (DNS) (ドメイン・ネーム・システム (DNS) での RSA/MD5 KEY と SIG)
2538		Storing Certificates in the Domain Name System (DNS) (ドメイン・ネーム・システム (DNS) での証明書の格納)
2539		Storage of Diffie-Hellman Keys in the Domain Name System (DNS) (ドメイン・ネーム・サービス (DNS) での Diffie-Hellman キーの格納)
2541		DNS Security Operational Considerations (DNS セキュリティの操作上の考慮事項)
2553		Basic Socket Interface Extensions for IPv6 (IPv6 用基本ソケット・インタフェース拡張) (RFC 2133 の置き換え)
2671		Extension Mechanisms for DNS (EDNS0) (DNS の拡張メカニズム (EDNS0))

表 A-1: RFC 標準 (続き)

RFC	プロトコル	名前
2672		Non-Terminal DNS Name Redirection (非ターミナル DNS 名のリダイレクション)
2673		Binary Labels in the Domain Name System (ドメイン・ネーム・システムのバイナリ・ラベル)
2710		Multicast Listener Discovery (MLD) for IPv6 (IPv6 用 Multicast Listener Discovery (MLD)) (マルチキャスト・ルータは現在インプリメントされていない)
2741		Agent Extensibility (AgentX) Protocol Version 1 (Agent Extensibility (AgentX) プロトコル・バージョン 1) (RFC 2257 の置き換え)
2874		DNS Extensions to Support IPv6 Address Aggregation and Renumbering (IPv6 アドレス集約および再番号付けをサポートするための DNS 拡張)
2893		Transition Mechanisms for IPv6 Hosts and Routers (IPv6 ホストおよびルータの転送メカニズム) (片方向構成トンネルは現在インプリメントされていない) (RFC 1933 の置き換え)

A.2 RFC 以外の標準

Tru64 UNIX オペレーティング・システムは、RFC (Request for Comment) 以外の下記の標準をインプリメントしています。

- 4.3BSD および 4.4BSD の ソケット・インタフェース
- 4.3BSD inetd
- 4.3BSD lpd
- 4.3BSD netstat
- 4.3BSD ping
- 4.3BSD rcp
- 4.3BSD rexecd
- 4.3BSD rlogin
- 4.3BSD rmt
- 4.3BSD rsh
- 4.3BSD syslog

- 4.4BSD Sendmail V8.9.3
- UUCP Basic Networking Utilities (HoneyDanBer)
- X/Open Transport Interface (XTI)
- Sun Open Network Computing (ONC) 4.2
- オプションの `nrdisk` としてパッケージ化されている新しい `rdist` コマンド
- BSD Packet Data Compression (PPP 用)

UNIX および一般コンピュータの用語集

本用語一覧では、Tru64 UNIX のマニュアルで見られる主な用語を定義しています。大部分の用語は UNIX 環境に関連するものですが、それ以外にも、インターネット関連用語など共通な用語も含まれています。

記号

/

root を参照

.(ドット)

ユーザの作業ディレクトリを表す簡略表現。

作業ディレクトリ も参照

.. (ドット・ドット)

ユーザの作業ディレクトリのすぐ上にある親ディレクトリを表す簡略表現。

8 進数

8 をベース (基数) として使用する記数法。8 進法では 0 から 7 までの数字を使用し、それぞれの桁位置が 8 の累乗を表します。

A

API (アプリケーション・プログラム・インタフェース)

特定のプログラム (アプリケーション) またはコンピュータ・オペレーティング・システムを規定する方式であり、これを使用してアプリケーション・プログラムを記述すると、特定のプログラムまたはオペレーティング・システムに対して要求を出すことができます。

apropos

特定のワードまたは文字の文字列を含む参照ページ名および要約行を表示するコマンド。apropos コマンドは `man -k` コマンドと同じ働きです。

リファレンス・ページ, *man* も参照

ARP (アドレス解決プロトコル)

- 1) 上位のインターネット・アドレスを低位のインターネット・アドレス，つまり物理的なハードウェア・アドレスに動的に結びつけることができるインターネット (TCP/IP) プロトコル。ARP を使用できるのは，単一の物理的ネットワーク内およびハードウェア・ブロードキャスト機能を備えたネットワークの場合だけです。
- 2) ローカル・エリア・ネットワーク (LAN) 上にあるインターネット・アドレス，ベースバンド・アダプタ・アドレス，およびトークンリング・アダプタ・アドレス間で，動的にマッピングするインターネット (TCP/IP) プロトコル。

ASCII

128 文字 (制御およびグラフィック文字を含む) を定義する標準文字セット。ASCII (情報交換用米国情報コード) が 7-ビットのバイナリ値をそれぞれの字，数字，および選択された制御文字に割り当てています。ASCII ファイルとテキスト・ファイル という用語は区別なく使用されます。

ATM (非同期転送モード)

セル交換を採用した 25 M/bps ~ 622 M/bps のネットワーク標準。接続を目的として情報交換機能を提供する全二重式のノード間通信回線のことです。

attribute-value pair (属性値ペア)

ソフトウェア製品キットの主要ファイルにおける，キットの単一属性の名前および値を特定する行。キットを `kit` コマンドでキットを作成する方法と，`setld` ユーティリティでインストールする方法を制御します。

awk

`awk` プログラミング言語で記述されたプログラムを実行するコマンド。`awk` プログラムはパターンとパターン読み込み時に実行される対応するアクションの連続で構成されます。`awk` ユーティリティはより強力なツールであり，`grep` または `sed` のいずれよりも，パターン照合およびテキスト操作に適しています。

grep , *sed* も参照

B

Berkeley Internet Name Domain

BIND (Berkeley Internet Name Domain) を参照

Berkeley Software Distribution

BSD (Berkeley Software Distribution) を参照

Berkeley UNIX

BSD (Berkeley Software Distribution) を参照

/bin directory (/bin ディレクトリ)

実行可能なプログラムとスクリプトを含むディレクトリ。たとえば，
/usr/bin ディレクトリには非特権ユーザが実行できるプログラムが，そ
して /sbin ディレクトリには特権ユーザのみが実行できるプログラムが含
まれます。

バイナリ・ファイル，パス，スクリプトも参照

BIND (Berkeley Internet Name Domain)

インターネット・ネットワークで使用できるネーム・サービス。

bit bucket (ビット・バケット)

取り出す可能性のないデータを保管する場所の名称。しばしば，/dev/null
として空デバイスを指すのに使用されます。

Bourne シェル

Steve Bourne が最初に開発したコマンド・インタプリタおよび解釈済みプ
ログラミング言語。

シェル も参照

break 文

プログラミング言語で，プログラムが現在の制御構造 (case 文または for ルー
プなど) から直ちに抜けるようにする文。break 文が頻繁に使用されるのは，
プログラムされた回数分反復を行う前に，ループの実行を終了する場合です。

BSD (Berkeley Software Distribution)

University of California at Berkeley の Computer System Research Group に
よる UNIX ソフトウェア・リリースであり，Tru64 UNIX オペレーティン
グ・システムの一部機能の基礎になっています。

BSD ソケット・インタフェース

アプリケーションに対して提供される一種のトランスポート層インタフェー
スであり，これを使用して単一システム上の関連のない2つのプロセス間
で，または接続された複数のシステム上でプロセス間通信を実行します。こ

のプロセス間通信機能を使うと、プログラムはその他のプログラム、プロトコル、およびデバイスとの通信にソケットを使用できます。

built-in

シェル内に組み込まれるコマンドであり、built-in の逆のコマンドの場合は、個別に実行可能な独立ファイルとなり、シェルによって呼び出されます。

C

C

構造化された手続き型プログラミング言語であり、オペレーティング・システムとアプリケーションの双方に広範に使用されるとともに、大学等で広く学ばれています。Tru64 UNIX オペレーティング・システムは C で記述されています。C は、Portable Operating System Interface (POSIX) の一部として標準化されています。

C++、コンパイラ、*Java* も参照

C++

大規模アプリケーション・プログラムの作成に最適な言語として現在一般的にみなされているオブジェクト指向プログラミング言語。C++ (*C* プラス・プラス と発音) は C プログラミング言語のスーパーセットです。

コンパイラ、*Java* も参照

call by reference (参照渡しコール)

プログラミング言語で、実値ではなくデータのアドレスを提供することにより、引数をサブルーチン、関数またはプロシージャのいずれかに渡す方式。

call by value (値渡しコール) も参照

call by value (値渡しコール)

プログラミング言語で、データの実値を提供することにより、引数をサブルーチン、関数またはプロシージャのいずれかに渡す方式。

call by reference (参照渡しコール) も参照

CAM (Common Access Method)

デバイス・ドライバと Host Bus Adapters (ホスト・バス・アダプタ) 間のソフトウェア・インタフェースを定義する ANSI 規格。また、SCSI 周辺機器がホスト・プロセッサに接続されていることを意味することもあります。

SCSI 小型コンピュータ用周辺機器インタフェース も参照

CAM Control Block (CAM 制御ブロック)

CCB (CAM Control Block (CAM 制御ブロック)) を参照

case insensitive (大文字小文字を区別しない)

大文字と小文字の区別ができないことです。大文字小文字を区別しない (case-insensitive) デバイスまたはプログラムは、A と a を同一文字として認識します。

大文字小文字を区別する も参照

case 文

プログラミング言語で、引数の評価に応じて使用可能な複数のパスから選ぶことができる制御構造。

cbreak mode (c ブレーク・モード)

プロセスがタイプされたとおりに入力を読み込むことができる端末ドライバ・オペレーション・モード。このモードには、文字、モード、および行編集入力機能がありません。

CCB (CAM Control Block (CAM 制御ブロック))

SCSI インタフェース・モジュール (SIM) ごとに関数の実行を制御するために、SCSI 周辺機器ドライバが XPT トランスポート・レベルに対して提供するデータ構造。

CDB (Command Description Block)

特定の演算に対応した SCSI 演算コード、パラメータ、および制御ビットを含むデータ構造。

CDE (Common Desktop Environment)

Tru64 UNIX オペレーティング・システムと対話するためのグラフィカル・ユーザ・インタフェース。CDE インタフェースは共同開発され、業界標準に基づいています。例として、X Consortium の X Window システムおよび Open Software Foundation の Motif インタフェースが挙げられます。

clist (c リスト)

BSD タイプの端末ドライバが、受信するデータまたは送信するデータを格納するために使用するデータ構造。

STREAMS も参照

Command Description Block

CDB (Command Description Block) を参照

Common Access Method

CAM (Common Access Method) を参照

Common Desktop Environment

CDE (Common Desktop Environment) を参照

cooked mode (クック・モード)

入ってくるデータをドライバが解釈する際のデバイス・ドライバの状況。たとえば、cooked mode (クック・モード) で動作している UNIX 端末ドライバは、端末からのリターン文字を行送り文字に翻訳してシステムに送ります。
raw モード も参照

cron

crontab ファイルの命令に従って、指定した回数と日付でコマンドを実行するデーモン。
デーモン も参照

crontab ファイル

特定のコマンドで日付と回数を指定するファイルが実行されます。*cron* デーモンが、指定した間隔で *crontab* ファイルを確認して、指定した日付と回数で表示されたコマンドを実行します。

csh

C シェルを呼び出すコマンド。
C シェル, シェル も参照

C シェル

University of California at Berkeley で開発されたコマンド・インタプリタおよび解釈済みプログラミング言語。その構造の多くが相当する C 言語構造に類似しているため、このように命名されています。
シェル も参照

D

Dataless Management Services

DMS (Dataless Management Services) を参照

dbxd

dbx プログラムを呼び出すコマンドであり、これを使用して開発者は開発中にその他のプログラムをデバッグするのに役立ってます。

DCE (分散コンピューティング環境)

分散コンピュータの業界標準であり、共通の命名、セキュリティ、時間同期化、システム可用性、データへのアクセス、およびシステム管理といった一定のグローバル・プロパティを共有するサービスのユニフォーム・セットを定義します。DCE によって、異機種システム上でアプリケーションとデータが同時に動作できます。

DFS (分散ファイル・システム)

全体的に分散化したファイルを一元管理するシステムを提供する分散 DCE アプリケーション。このファイル・システムのもと、DFS ファイルには現在ファイルを格納しているサーバによらず、同一名を使用して DCE DFS マシンからアクセスすることができます。

DHCP (Dynamic Host Configuration Protocol)

このインターネット (TCP/IP) プロトコルによって、使用可能なアドレスのプールからインターネット・アドレスをネットワーク上のクライアントに自動割り当てできます。アドレスの割り当ては、携帯型コンピュータなどのクライアント・システムがネットワークに接続されると自動的に実行されます。

Distributed Computing Environment (分散コンピューティング環境)

DCE (分散コンピューティング環境) を参照

Distributed File System (分散ファイル・システム)

DFS (分散ファイル・システム) を参照

DMS (Dataless Management Services)

HPwhereby サーバ・コンピュータ・システムが提供するサービスでは、サーバに接続されたクライアント・コンピュータ・システムの `root`、`/usr`、および `/var` ファイル・システムを、ローカル・エリア・ネットワーク (LAN) を通して保持します。

DMS クライアント

システム・ディスクがクライアント自体ではなく DMS サーバに物理的に接続されているコンピュータ・システムであり、クライアントはネットワークを経由してそこにアクセスします。

DMS 領域

予約済みのディスク領域であり、DMS サーバに物理的に接続されており、そこには DMS ルート領域の複数コピー、つまり DMS クライアントにつき 1 つずつ含まれています。

dupatch

ユーティリティの一種であり、インストール、削除、および Tru64 UNIX や TruCluster 製品のパッチを管理するパッチ・キットに含まれます。このユーティリティは、Tru64 UNIX パッチ・キットのインストールが問題なく完了すると、システム上にインストールされて残ります。

Dynamic Host Configuration Protocol

DHCP (Dynamic Host Configuration Protocol) を参照

E

ed エディタ

テキスト・ファイルのコンテンツを修正するための行指向プログラム。プログラムはユーザからコマンドを受け取ることによって動作します。たとえば、`s/Unix/UNIX/g` コマンドを出すと、エディタは現在の行にある “Unix” という文字列のそれぞれを “UNIX.” に置換します。

EGP (External Gateway Protocol)

個別のネットワークがインターネット・バックボーンと通信できるようにするルーティング・プロトコルのタイプ。

インターネット も参照

Emacs

Free Software Foundation が開発したテキスト・エディタであり、すべての UNIX システムが使用できますが、Berkeley UNIX または System V の標準ではありません。これは Tru64 UNIX オペレーティング・システムと同梱されています。

email (電子メール)

記述したメッセージを他のユーザとネットワーク上で交換可能にするシステム。

EOF (ファイルの終わり)

1) プログラムがファイルを読み込んでデータ・ファイルの終わりに到達したことを示す状況。

2) 磁気テープに記述された文字の特定の順序。

ファイル・マーク も参照

/etc

catchall ディレクトリであり、通常いろいろなシステム・データ・ファイルが含まれます (例、termcap、つまり端末機能データベースなど)。

Ethernet

異種コンピュータ、情報処理製品、および事務機器同士を内部接続するローカル通信ネットワークの通信概念。ベースバンドが 10 M バイト/秒のローカル・エリア・ネットワーク (LAN) であり、衝突検出機能付キャリア検知多重アクセス (CSMA/CD) を採用しています。ネットワークを通して、複数の局が事前調整なくして自在にメディアにアクセスできます。また、キャリア検知と遅延、さらに検出と送信を実施することによって送信権の争奪を回避します。

External Gateway Protocol

EGP (External Gateway Protocol) を参照

G

gid, GID

グループ ID (GID) を参照

globbing (グロビング)

ワイルドカードによるファイル名展開というシェルのプロセスを指す UNIX 用語であり、リテラル・ファイル名のリストを作成してからシェルがそれをコマンドに渡します。C シェルによってユーザのデフォルトによる globbing (グロビング) が不可能になります。Bourne, Korn, および POSIX シェルでは、globbing (グロビング) を望まない場合は、ユーザはファイル名のメタキャラクタに引用符を付けるか、または、メタキャラクタの使用を避ける必要があります。

grep

grep プログラムを呼び出すコマンドであり、それを使用して指定したファイルから指定したパターンに適合する文字が含まれる行を探索してから、その適合した行を標準出力に記述します。名前は、Global Regular Expression Printer を意味しています。

正規表現 も参照

H

hashed passwd database (ハッシュド passwd データベース)

passwd ファイルのコンテンツを含むインデックス付きデータベース。インデックス付きデータベースは情報を取り出すのに必要な探索時間を最小限にします。

HBA (Host Bus Adapter (ホスト・バス・アダプタ))

システム・メモリと SCSI (Small Computer System Interface) バス間のインタフェースを提供するハードウェアおよびマイクロコード。

history

C シェルおよび Korn シェルにおけるユーザのヒストリ・リストを表示するコマンド。

\$HOME

ユーザのホーム・ディレクトリの絶対パス名を含む環境変数。

\$home , 環境変数 も参照

\$home

ユーザのホーム・ディレクトリの絶対パス名を含むプロセス変数。

\$HOME , プロセス変数 も参照

Host Bus Adapter (ホスト・バス・アダプタ)

HBA (Host Bus Adapter (ホスト・バス・アダプタ)) を参照

HTML (ハイパーテキスト・マークアップ言語)

World Wide Web ブラウザ上での表示を意図してファイルに挿入されたコーディング (マークアップ) のことであり、ブラウザに Web ページのワードをどのように表示するかを伝えます。マークアップはタグを使用して表されますが、そのコマンド・ワードは山カッコで囲まれます。たとえば、<P> というタグは新しいパラグラフを作成します。また、<TABLE> というタグはテーブルのフォーマットを開始します。World Wide Web Consortium (W3C) は HTML の標準化を進めていますが、Netscape も Microsoft のブラウザも、現在一部の機能が異なったままインプリメントされており、かつ非標準拡張子を採用しています。

PDF ファイル も参照

I18N

国際化 を参照

ICMP (Internet Control Message Protocol)

インターネット・プロトコル群のうちのホスト・ツー・ホストのプロトコルのことであり、Internet Protocol (IP) のエラーと操作を制御します。

IP (インターネット・プロトコル) も参照

#include

C 言語プリコンパイラ指示文であり、名前付きファイルの挿入方法をコンパイルされたファイルに指定します。挿入されるファイルは標準ヘッダ・ファイル (名前を山カッコで囲んで表示) または C 言語コードを含むそれ以外のファイル (名前を二重引用符で囲んで表示) です。たとえば、

```
#include <header_file.h>
#include "myfile.c"
```

名前が山カッコ (<>) で囲まれているヘッダ・ファイルの絶対パス名は、
/usr/include/file.h です。

インクルード・ファイル も参照

init

ルート・ファイル・システムのマウント後のブート・プロシージャの最終ステップとして、UNIX システムが出すコマンド。init プログラムはプロセスを作成して制御することによってシステムを初期化します。また、そのプロセスは inittab ファイルで定義されています。

init プロセス

システム・アドミニストレーション・タスクを実行するシステムが作成するプロセス。例、ログイン・プロセスの生成およびマルチユーザからシングルユーザ・モードへの順序立ったシャットダウン処理など。

inode

オペレーティング・システム内の個々のファイルを表す内部構造。各ファイルには inode が 1 つあります。inode には、ファイルのノード、タイプ、所有者、およびロケーションが含まれます。inode のテーブルはファイル・システムの開始地点近くに格納されます。

inode 番号

ファイル・システム内の特定の inode ファイルを指定する番号。

Internet Control Message Protocol

ICMP (Internet Control Message Protocol) を参照

IP (インターネット・プロトコル)

インターネット・プロトコル群のネットワーク層プロトコルであり，コネクションレス，つまり最適パケット伝送サービスの基盤を提供します。IP には Internet Control Message Protocol (ICMP) が必須部分として含まれています。インターネット・プロトコル群は，IP が 2 種類のもっとも基本的なプロトコルの 1 つであることから，TCP/IP と呼ばれます。

IP ゲートウェイ

IP ルータ を参照

IP ルータ

2 つ以上のインターネット・ネットワークを接続するホスト。IP ルータはルータが組み込まれているネットワーク上のホストすべてに到達する道順を把握しています。IP ゲートウェイとも呼ばれます。

ISO (国際標準化機構)

89 ヶ国の国家標準化組織で構成される国際母体。ISO はネットワーキング・ソフトウェアを含む多岐にわたる商品やサービスの規格を發布します。

J

Java

Sun Microsystems によって開発され，C++ に基づく，インターネットなどのネットワークでプログラム・オブジェクトの配布のために最適化されたプログラム言語。Java は C++ より単純で学習が容易であり，またその他の利点を持っています。

Java アプレットは Java で書かれた，ひとつのウィンドウやプルダウン・メニューからなる短いプログラムです。

JavaScript

Netscape により開発されたインタプリタ型プログラミング (スクリプト) 言語。機能的には Microsoft の Visual Basic，Sun の Tcl，UNIX から派生した Perl，IBM の REXX などに類似します。一般的に，スクリプト言語は C や C++ などの高度に構造化されコンパイラ型言語よりも，容易かつ迅速にコードすることができます。スクリプト言語は一般にコンパイラ型言語より処理に時間がかかりますが，短いプログラムを作成するには便利です。

K

kdbx

kdbx プログラム，対話型のクラッシュ分析，カーネル・デバッグ・ツールを呼び出すコマンド。 **kdbx** プログラムは **dbx** デバッガのフロントエンドとして機能します。

kdebug プログラム

実行するカーネルを指定するプログラム。

kill

- 1) プロセスの強制的な停止。通常は Ctrl-c を押すことで，ユーザはフォアグラウンド・プロセスを停止できます。
- 2) ユーザがバックグラウンド・プロセスや中断プロセスを終わらせるために実行する Tru64 UNIX コマンド。スーパーユーザはこのコマンドを実行してシステム上のどのプロセスでも停止できます。

Korn シェル

David Korn によって開発されたコマンド・インタプリタとインタプリタ型プログラム言語。 **Korn** シェル (**ksh**) は意味論的には **Bourne** シェルの拡張されたバージョンであり，ジョブ管理やコマンド・ヒストリ再呼出しなどの拡張機能を実装するコマンドを含みます。 **POSIX** シェルは **Korn** シェルのスーパーセットです。

シェル も参照

L

L10N

ローカリゼーション を参照

LAN (ローカル・エリア・ネットワーク)

情報処理装置の接続に最適化された高速の通信チャネルを提供する，限定された物理的距離で稼働するデバイス通信システム。

LAT (ローカル・エリア転送)

ローカル・エリア・ネットワーク (LAN) 上で，ホスト・コンピュータ・システムと端末，PC，プリンタ，モデム，その他の装置が接続されたターミナル・サーバとの間の通信をサポートするプロトコル。

LAN (ローカル・エリア・ネットワーク) も参照

LDAP (Lightweight Directory Access Protocol)

TCP/IP 上で動作する，インターネット標準のディレクトリ・サービス・プロトコル。LDAP サーバはディレクトリ内のエントリを管理し，その情報をネットワーク上のユーザやアプリケーションから利用できるようにします。この情報は，個々のユーザを識別し認証するために使用する，ユーザ情報の中心的なレポジトリとして使用できます。このような使い方では，LDAP サーバは NIS (Network Information Services) と似ています。

LAN (ローカル・エリア・ネットワーク) も参照

lex

入力を意義素と呼ばれる意味 (シンボル) の単位に整理するプログラムを生成するためのプログラムである，字句解析ルーチン生成システムを呼び出すコマンド。

字句解析プログラム，パーサ，*yacc* (*Yet Another Compiler-Compiler*) も参照

LUN (論理ユニット番号)

SCSI ドライバが識別できるように SCSI デバイスに割り当てられた番号。

M

MAKDEV

Tru64 UNIX システムのデバイスのためにデバイス特殊ファイルを作成するスクリプト。このスクリプトは */dev* ディレクトリにあります。

make

特定のアプリケーションを作成するソースファイルが作成されるターゲット・ファイルより新しいかテストをして，プログラムとアプリケーションを構築するツール。もしソースや中間形式のファイルがターゲット・ファイルより新しいなら，*make* は一定の規則に従ってターゲット・ファイルを再構築するために必要な処理を行います。規則はデフォルトを使用するか，必要な手順を明示的に指定できます。

makefile

make ツールが使用する，仕様を含むファイル。*makefile* にはターゲット・プログラムの名前を指定し，その構築のための規則が記述されます。

make も参照

man

オンラインでリファレンス・ページを表示するコマンド。*manual* の略です。

apropos , リファレンス・ページ も参照

manpage , マニュアル・ページ
リファレンス・ページ を参照

MANPATH

`man` , `catman` , `xman` コマンドによって使用される , デフォルトのディレクトリ検索パスを規定する環境変数。

探索パス も参照

MFS (メモリ・ファイル・システム)

メモリだけに存在する UFS ファイル・システム。恒久的なデータまたはファイル構造はディスクに出力されません。MFS は読み込み/書き込み性能を改善しますが、一時的なキャッシュでしかありません。MFS の内容はリブート , アンマウント操作 , 停電時などには失われます。

MIB (管理情報ベース)

管理情報ベースはネットワーク管理に関連する一連のデータ要素を定義します。これらの多くはインターネット・ソサエティのインターネット・エンジニアリング・タスクフォース作業グループが作成したRFC により標準化されています。

Motif

OSF/Motif を参照

mount

ファイル・システムを利用可能にするコマンド。

アンマウント も参照

N

NetRAIN (Redundant Array of Network Adaptors : ネットワーク・アダプタの冗長配列)

特定の種類のネットワーク接続性障害を防止するための機構を提供するインタフェース。NetRAIN は同じ LAN セグメントの複数のネットワーク・インタフェースを NetRAIN セットと呼ばれる単一の仮想インタフェースに統合します。他のネットワーク・インタフェースがアイドル状態でも、セットの中の 1 つのネットワーク・インタフェースは常にアクティブです。もしアクティブなインタフェースに障害が発生すれば、アイドル状態の

セットのメンバの 1 つが調整可能なフェイルオーバー時間周期内に同じ IP アドレスでオンラインになります。

NFS (ネットワーク・ファイル・システム)

システム (サーバ) が、ネットワーク経由でその他のシステム (クライアント) にファイル・システムをマウントさせて利用可能にするサービス。クライアントが NFS ファイル・システムをマウントすれば、クライアントのユーザはローカルのものと同様にファイル・システムを使用できます。

NFS マウントされた

マウントされるシステムについて、物理的に接続されている (ローカル) ファイル・システムに対して、NFS によりネットワーク経由でマウントされるファイル・システムを指します。

NFS (ネットワーク・ファイル・システム) も参照

NIS (Network Information Service)

ローカル・エリア・ネットワーク (LAN) 上の情報を共有するための、分散データ・ルックアップ・サービス。NIS を使用すると、ネットワーク環境でのデータベース情報を適度に分散させることができます。NIS は以前、イエロー・ページと呼ばれていました。

NFS (ネットワーク・ファイル・システム) も参照

NUMA (不均等メモリ・アクセス)

マルチプロセッサ・システム用の 2 つのアーキテクチャのうちの 1 つ。もう 1 つは、シンメトリック・マルチプロセッサ (SMP) です。NUMA アーキテクチャは、1 つのインタコネクト (すべてのシステム・リソースにリンクしているバスまたはスイッチ) がある従来の SMP システムの欠点を改善します。NUMA を使用すると、システム・スイッチやバスが性能上のボトルネックになることなく、多数の CPU を組み込んでシステムを拡張することができます。

nroff

`nroff` プログラムを呼び出すコマンド。テキスト・フォーマットの *roff* ファミリのメンバ。`nroff` プログラムはターミナルやプリンタなどの文字セル装置上で、表示または印刷に適した ASCII 出力を生成します。

O

OLAR (オンラインでの追加および交換)

オペレーティング・システムのサービスやアプリケーションを動作させたまま、容量の拡大、構成要素のアップグレード、および障害のある構成要素の交換を行うために使用されるシステム管理プロセス。この機能(「ホット・スワップ」と呼ばれることもある)を使用すると、予定されたメンテナンスおよび予定外のメンテナンスを行う際にシステムが動作している時間を長くでき、可用性が向上するという利点があります。

OSF (オープン・ソフトウェア財団)

共通の機能に基づいた、幅広い互換性を持つ UNIX システムを開発して、市場に出す目的で組織されたソフトウェア・ベンダのコンソーシアム。

OSF/Motif

当初オープン・ソフトウェア財団により、後に Open Group によって開発され、ライセンスされたグラフィカル・ユーザ・インタフェース。OSF/Motif は X ウィンドウ・システムに基づいています。 *Motif* と呼ばれることもあります。

OSI (開放型システム間相互接続)

国際標準化機構によって開発された一連の国際規格。OSI の目標は、異なるベンダーのコンピュータ・システムの相互接続を可能にすることです。

P

PALcode (特権を与えられたアーキテクチャ・ライブラリ)

特定の Alpha オペレーティング・システム実装に固有の一連のサブルーチン。これらのサブルーチンは文脈切り替え割り込み、例外、メモリ管理のためのオペレーティング・システム・プリミティブを提供します。

passwd

- 1) ユーザが自分のログイン・パスワードを変更するコマンド。
- 2) 利用者パスワードと関連づけられた情報が保存される UNIX ファイル。ファイルのパス名は `/etc/passwd` です。

\$PATH

ユーザのコマンドの検索パスを持つ環境変数。\$PATH 変数でのディレクトリ名はコロンで区切られます。

\$path も参照

\$path

ユーザのコマンドの探索パスを持つプロセス変数。 *\$path* 変数でのディレクトリ名はスペースで区切られます。

\$PATH も参照

Perl

World Wide Web で商品を注文するためのフォームなどに使用される CGI (コモン・ゲートウェイ・インタフェース) プログラムの開発にしばしば使われる多用途のスクリプト言語。 Perl は構文的に C 言語 に似ており、**sed** と **awk** のような多用される UNIX の機能を多く含んでいます。 Perl は実行の直前に C コードまたはクロス・プラットフォームのバイト・コードにコンパイルされるインタプリタ型の言語です。 コンパイルされれば、Perl プログラムは完全にコンパイル済みの C 言語プログラムと同じくらい速く実行できます。

スクリプト も参照

PDF ファイル

Adobe Acrobat Reader によって使用される、文書を表示し、出力するための容易な方法を提供するファイル形式。 Adobe の PostScript の次世代フォーマットとして、PDF ファイルは特に CD-ROM とインターネットを経由して文書を配布するための標準的な手段になりました。 ほとんどの Tru64 UNIX 文書は PDF と **HTML** 形式で提供されます。 Acrobat Reader はバージョン 4.0E 以後の Tru64 UNIX Documentation CD-ROM で提供されます。 Adobe Web サイトでも同様に入手可能です。

<http://www.adobe.com/>

pid , PID

プロセス ID (*PID*) を参照

POSIX (ポータブル・オペレーティング・システム・インタフェース)

米国電気電子学会 (IEEE) の POSIX 作業グループによって提案された標準規格のコレクション。 POSIX 規格はアプリケーションのソース移植性をサポートするためのシステム・インタフェースを定義します。

C , *SVID* (*System V* インタフェース定義) も参照

POSIX シェル

POSIX 規格に従うシェル。 POSIX シェル (**sh**) は **Korn** シェルのサブセットです。

シェル も参照

PostScript

Adobe Systems によって開発された，活字組み文書や表示用文書の書式を指定するための言語。Encapsulated PostScript ファイルは，PostScript ファイルに別の PostScript ファイルを埋め込む規格に基づくファイルです。

PDF ファイル も参照

PPP (2 地点間プロトコル)

非同期のシリアル・ラインに関して IP データグラムをカプセル化して転送する転送ライン・プロトコル。PPP は SLIP より効率的です。

SLIP (シリアル・ライン・インターネット・プロトコル) も参照

printcap データベース

システムに認識できるすべてのプリンタの詳細を含むファイル (/etc/printcap)。

pty

擬似ターミナル を参照

pwd

ユーザの作業ディレクトリの絶対パス名を表示させるコマンド。

作業ディレクトリ も参照

R

RAID (Redundant Array of Independent Disks : 独立したディスクの冗長配列)

同じデータを冗長的に，つまり複数のハード・ディスク上の別の場所に保存する方法。データを複数のディスク上に置くことで，I/O 処理をバランスよく重複させることができ，性能を改善します。

RAID 技術は Logical Storage Manager (LSM) またはハードウェアに基づく RAID サブシステムを使用するストレージ機器構成で，高性能と高度なデータ可用性を実現するために多く使用されます。

4つの主要な RAID レベルは，データまたはディスクストライピングとして知られている RAID 0，データまたはディスクミラーリングとして知られている RAID 1，そしてパリティ RAID の一種である RAID 3 と RAID 5 です。

raw ソケット

内部のネットワーク・プロトコルとインタフェースに特権を与えられたユーザ・アクセスを提供するソケット。これらのソケットは通常のインタフェースでは利用できないプロトコル機能を利用する際や、ハードウェア・インタフェースを使用して通信するために使用されます。

raw モード

ドライバがそれを通してデータを解釈しないデバイス・ドライバの条件。たとえば、raw モードで稼働している UNIX 端末ドライバは端末から直接システムに Return 文字を渡します。

cooked mode (クック・モード) も参照

rc

コンピュータを起動する過程を管理するコマンド・スクリプトを含むファイルの名前の要素。rc 文字は、BSD メール・ユーティリティ *.mailrc* や、Motif ウィンドウ・マネージャ *.mwmrc* などの、ユーザによってカスタマイズされたスタートアップ情報を含むファイル名でも使用されます。

RCS (リビジョン管理システム)

特定のファイルのどの改訂版でも入手できるように、プログラムとマニュアル・ソース・ファイルを管理するための一連のプログラム。ファイルへの改訂は、すべてのバージョンの完全なコピーではなく、元のバージョンに適用された一連の逐次的変更 (デルタ) として保存されます。一度に 1 人のユーザだけが特定のファイルを変更できるように、システムはロック機構を提供します。

SCCS (Source Code Control System) : ソース・コード管理システム) も参照

RCS ファイル

元のファイルのテキストと適用された変更のリストを含むリビジョン管理システム (RCS) ライブラリに保存されたファイル。

RCS ライブラリ

リビジョン管理システム (RCS) ファイルが保存されるディレクトリ。

Redundant Array of Network Adaptors (ネットワーク・アダプタの冗長配列)

NetRAIN (Redundant Array of Network Adaptors) : ネットワーク・アダプタの冗長配列) を参照

Request For Comments

RFC (Request For Comments) を参照

RFC (Request For Comments)

インターネットの構成要素を記述する形式仕様または文書。RFC は技術者のグループによって立案され、他の技術者によって調べられるために提出されます。一度承認を受ければ、RFC には番号が付けられ、決定的な仕様書とみなされます。番号を付けられた RFC は変更できませんが、次の RFC が以前の RFC のすべてまたは一部に取って代わるか、またはさらに詳述を加えることができます。

RISC (縮小命令セット演算)

大きくて、さまざまな複雑な命令のセットの代わりに、限定された単純な命令のセットに基づくコンピュータ・アーキテクチャ。

RIS (リモート・インストール・サービス)

ローカルにマウントされた配布メディアを使用する代わりに、ネットワーク経由でソフトウェア・キットをインストールするためのユーティリティ。

RIS クライアント

サーバの RIS 領域に保存されたキットにアクセスして、ネットワーク経由でソフトウェアをインストールする許可を持つコンピュータ・システム。

RIS サーバ

その他のコンピュータに、インストールするソフトウェア・キットを提供するコンピュータ・システム。ソフトウェアはサーバに属するディスクに保存され、RIS クライアントによってネットワーク経由でアクセスされます。

RIS 領域

インストール可能なソフトウェア・キットが保存されている1つ以上の製品環境を含む、物理的にRIS サーバに接続されている予約されたディスク領域。

root

- 1) スーパユーザ (システム管理者) のログイン名。
- 2) この名前は UNIX システムのツリー型ファイル構造体の最上位のディレクトリに適用されます。つまり、絶対パス名の始まりでもあります。root ディレクトリは先頭スラッシュ (/) を持つパス名で表されます。root ディレクトリ自体への参照が 1 つのスラッシュから成り立ちます。

パス名 も参照

root ディレクトリ

root を参照

root ファイル・システム

その他のすべてのファイル・システムがマウントされる基本的なファイル・システム。root ファイル・システムはシステムのその他の部分を実行するためのオペレーティング・システム・ファイルを含んでいます。

root ログイン

root を参照

S

SCCS (Source Code Control System : ソース・コード管理システム)

特定ファイルのどのようなリビジョンでも入手できるように、プログラムとマニュアル・ソース・ファイルを管理するための一連のプログラム。ファイルへのリビジョンは、すべてのバージョンの完全なコピーではなく、元のバージョンに適用された一連の逐次的変更(デルタ)として保存されます。一度に1人のユーザだけが特定のファイルを変更できるように、システムはロック機構を提供します。

RCS (リビジョン管理システム) も参照

SCCS ライブラリ

ソース・コード管理システムの *s* ファイルと *p* ファイルが保存されるディレクトリ。

SCP (サブセット管理プログラム)

ソフトウェアのインストールを管理するためにソフトウェア・キットに含まれるファイル。SCP はキットの開発者によって書かれ、そしてキットのインストール中に **setld** ユーティリティによって呼び出されます。システム管理プログラムファイルは拡張子 *.scp* を持っています。

SCSI インタフェース・モジュール

SIM (*SCSI* インタフェース・モジュール) を参照

SCSI 小型コンピュータ用周辺機器インタフェース)

パーソナル・コンピュータ、小規模のマルチユーザ・システム、ワークステーションなどのような小規模のシステムのための工業規格バス。SCSI ベースの装置は、複数の装置を同じバスで直列に構成することができます。SCSI は“スカジー”と発音されます。

sed

標準的なストリーム・エディタである `sed` ユーティリティを呼び出すコマンド。`sed` エディタは1つ以上のテキスト・ファイルを読み取り、一連の編集コマンドを含むスクリプトに従って変更を加え、標準出力に結果を書き込みます。

setld

ソフトウェア・サブセットをインストール、管理、更新、削除するためのユーティリティ。
サブセット も参照

sh

`passwd` ファイルのユーザ設定に基づいて **Bourne** シェル、または**POSIX** シェルを呼び出すコマンド。

SIM (SCSI インタフェース・モジュール)

SCSI コマンドを実行するため、XPT トランスポート層を通して送られる CAM 制御ブロックを承認するよう設計されたサブプログラム。

SLIP (シリアル・ライン・インターネット・プロトコル)

非同期のシリアル・ライン上で IP データグラムをカプセル化して転送するライン・プロトコル。SLIP は PPP ほど効率的ではありません。
PPP (2 地点間プロトコル) も参照

SMTP (シンプル・メール転送プロトコル)

電子メールを交換するためのインターネット規格プロトコル。

SNMP (単一ネットワーク管理プロトコル)

ネットワーク管理情報を交換するためのインターネット規格プロトコル。

stdin

標準入力 を参照

store-and-forward

次の中間ホストへの転送を始める前に、完了した転送が1つの中間ホストに引き渡されるネットワーク接続の一種。

STREAMS

デバイス・ドライバの実装とネットワーキング・プロトコル・スタックをサポートするカーネル機構。

clist (c リスト), STREAMS フレームワーク も参照

STREAMS フレームワーク

文字入出力のために、カーネル内と、カーネルとユーザ・レベルの間でインタフェース標準規格を定義する STREAMS コンポーネント。これらのコンポーネントには関数、ユーティリティ・ルーチン、カーネル機能、データ構造が含まれます。

stty

ユーザ端末の特定の属性を設定またはレポートするコマンド。

su

スーパーユーザ を参照

SVID (System V インタフェース定義)

System V で、サブルーチン・コール、システム・コール、コマンド、ユーティリティ、サービスを定義する仕様。

POSIX (ポータブル・オペレーティング・システム・インタフェース) も参照

SVVS (System V 検証スイート)

System V インタフェース定義への遵守をテストする一連のプログラム。

sync

システムが停止する前に、すべてのキャッシュされたディスク書き込み操作の完了を強制するコマンド。

System V

AT&T によって開発された UNIX システムのバージョン。

System V インタフェース定義

SVID (*System V* インタフェース定義) を参照

System V 検証スイート

SVVS (*System V* 検証スイート) を参照

T

tar ファイル

`tar` コマンドにより作成され、単一のファイルに複数のファイルを保存、復元するファイル。Tru64 UNIX CD-ROM に含まれているキット以外の Tru64 UNIX パッチ・キットが `tar` ファイルとして提供されます。

TCP (伝送制御プロトコル)

アプリケーションのために、信頼性が高く、全二重のコネクション指向のサービスを提供するインターネット・トランスポート層プロトコル。TCP がネットワークを通して情報を転送するために IP プロトコルを使用します。

TCP/IP

インターネット・プロトコル・スイートの 2 つの基本的なプロトコル、またはインターネット・プロトコル・スイートを示すためにしばしば使用される頭字語。TCP はデータの高信頼転送を提供し、他方 IP はデータグラムのかたちでネットワークを通してデータを送信します。

TCP (伝送制御プロトコル)、*IP* (インターネット・プロトコル) も参照

terminfo データベース

端末の種類と機能の記述を含むファイル。特定の端末が管理される方法を判定するためにシステムと X/Open Curses ライブラリ・ルーチンによって使用されます。

タームキャップ・データベース も参照

tty

端末の略語。

U

UBC (Unified Buffer Cache: ユニファイド・バッファ・キャッシュ)

従来型のファイル処理で読み取りおよび書き込みのために最近アクセスされたファイル・システム・データを一時的に保持し、マップされたファイル・セクションからページ・フォールトを保持することで、オペレーティング・システムとストレージ・ディスク間でレイヤとして動作するキャッシュ。

プロセスと UBC は限られた物理メモリを取り合うため、仮想記憶サブシステムはプロセスとファイル・システム要求に従って物理的なページを割り当てます。競合するメモリ資源の要求に応じるために、仮想記憶サブシステムは、最も古いページをスワップ空間に定期的書き込むことで、メモリ資源を再生利用します。高負荷の下では、すべてのプロセスがメモリの解放のために一時停止 (スワップアウト) することがあります。

VFS (Virtual File System : 仮想ファイル・システム) も参照

UDP (User Datagram Protocol : ユーザ・データグラム・プロトコル)

リモートのマシン上のアプリケーション・プログラムで相互にデータグラムを送るためのインターネット・プロトコル。UDP はデータグラムを配送するために IP を使用します。

uid , UID

ユーザ *ID (UID)* を参照

ULTRIX

Tru64 UNIX オペレーティング・システムの前身。Ultrix オペレーティング・システムは Digital Equipment Corporation によって開発され、VAX と RISC コンピュータ上で稼働するのに対し、HP Tru64 UNIX は Alpha システム上で稼働します。

umask

作成されるファイルのデフォルトのアクセス権を指定する 3 桁の、8 進法に基づく数列。umask コマンドはこの数を設定または変更します。

UNIX

元来、1960年代後期と1970年代初期に AT&T のベル研究所で開発され、その後カリフォルニア大学バークレー校、AT&T、Open Software Foundation (OSF)、その他の組織により発展したオペレーティング・システム。

UNIX-to-UNIX Copy Program

UUCP (UNIX-to-UNIX Copy Program) を参照

unlink

linkシステム・コールにより作成されたファイル間の接続を切断するために使われるシステム・コール。

URL (Uniform Resource Locator)

インターネットでアクセス可能なファイルまたはその他のリソースのアドレス。ファイルまたはリソースの種類はインターネット・アプリケーション・プロトコルに基づきます。たとえば HyperText Transfer Protocol (HTTP) を使用した場合、ファイルの種類は HTML ページ、画像ファイル、CGI アプリケーション、**Java** アプレットのようなプログラムなどです。このようなアドレスは以下ようになります。**http://tru64unix.compaq.co.jp** これは HP Tru64 UNIX Web サイトの URL です。

/usr

オペレーティング・システム，そしていくつかのアプリケーションのコンポーネントが保存される読み取り専用ファイル・システム。ユーザのホーム・ディレクトリが時々 /usr のサブディレクトリに置かれることがあります。

UUCP (UNIX-to-UNIX Copy Program)

ダイヤルアップ回線を使用してコンピュータを接続するための一連のプログラムとプロトコル。このプログラムは，ファイルのコピー，リモート・コンピュータへのログイン，そしてバイナリ・ファイルを転送のために 7 ビットの ASCII のデータ行にコード化するための機能を含みます。接続の容易さと低コストのため，UUCP は世界中で最も多用される情報ネットワークのひとつになりました。

UUCP ネットワーク

UUCP プログラムを使用して接続されたコンピュータの集合を指す用語。

V

VFS (Virtual File System : 仮想ファイル・システム)

ファイルが位置するファイル・システムにかかわらずファイルへの共通のアクセスを許可する統合インタフェース。

vi エディタ

フルスクリーン・テキスト・エディタ。vi エディタはモダル・エディタです。コマンド・モードで，カーソルの動き，テキスト削除などのコマンドが使えます。テキストをファイルに挿入するためには，ユーザはエディタを入力モードにするコマンドを入力すれば，その後エスケープキーが押されるまで，すべてのキー・ストロークが入力データと解釈されます。

全画面エディタ も参照

W

WEBES (Web-Based Enterprise Services)

HP により，多くのサービス・ツールに実装されている共通のアーキテクチャ。たとえば Compaq Analyze と Insight Manager は WEBES の一般的なコンポーネントを利用しつつ，独自の機能が追加されています。その他のサービス・ツールを同じマシンに加える際，共通の WEBES コンポーネントをそのまま利用することができます。それぞれの WEBES ベースの

サービス・ツールは、マシン上で継続的に実行される Director プロセスに機能性を加えます。

wordlist

1 語以上の単語から成り立つ C シェル変数。

WORM

記録可能コンパクトディスク (CD-R) などの、1 度だけ書き込むことができ、何度でも読み取ることができるストレージ媒体を指します。

X

X/Open トランスポート・インタフェース

XTI (*X/Open* トランスポート・インタフェース) を参照

XPT

SCSI 周辺機器ドライバが CAM (共通アクセス法) 機能を実行するためのソフトウェアのトランスポート層。

XTI (*X/Open* トランスポート・インタフェース)

プロトコル独立である、アプリケーションのためのトランスポート層インタフェース。XTI は、OSI モデルのためのトランスポート・サービス定義に基づく、トランスポート層インタフェース (TLI) に基づいた一連の C 言語関数から成り立ちます。

X ウィンドウ・システム

ネットワーク・ベースのウィンドウ化インタフェース。X ウィンドウ・システムは多くの主要なコンピュータ製造業者によって採用されています。

Y

yacc (Yet Another Compiler-Compiler)

パーサ (合理的な方法で入力を翻訳できるプログラム) を生成するためのプログラム。yacc の出力は C 言語プログラムです。yacc プログラムは通常 lex によって生成されたフロントエンドの出力を解釈するパーサを生成するために使用されます。

lex, パーサ も参照

あ

アーカイブ

- 1) プログラム，データ・ファイル，テキスト・ファイル，およびその他のタイプのファイルを安全確実に保存することを目的とします。
- 2) このようなファイルのレポジトリ。

アクセス権

ユーザが他のどのユーザまたはグループがファイルを読み取り，書き込み，実行できるか管理するためにファイルに指定する制約。3つの種類のアクセス権があります。ユーザに適用されるもの，ユーザのグループに適用されるもの，そして「その他」と呼ばれるその他の全員に適用されるものです。

アクセス権コード

アクセス権 を参照

アクセス権フィールド

アクセス権 を参照

アクティブ・ユーザ

XTI トランスポート接続の際に，接続を開始したトランスポート・ユーザを指します。

クライアント・プロセス，パッシブ・ユーザ，XTI (X/Open トランスポート・インタフェース) も参照

宛先変更

標準入力，標準出力と標準エラーの仮想ファイルが特定のコマンドの実行中に関連づけられるべき1つ以上の装置を指定すること。

アドレス解決プロトコル

ARP (アドレス解決プロトコル) を参照

アプリケーション

一定の有用な関数または関数のセットを実行するように設計されたプログラムまたはプログラムのセット。たとえば，ソース・コード制御システムはプログラムのソース・コードを管理するアプリケーションのことです。

アプリケーション・プログラム・インタフェース

API (アプリケーション・プログラム・インタフェース) を参照

アンマウント

特定のディレクトリ上に以前にマウントしたファイル・システムの除去をシステムへ通知すること。その特定のファイル・システムをマウントしたユーザ、またはスーパーユーザだけがそれをアンマウントすることができます。ファイル・システムは `umount` コマンドによりアンマウントされます。

い

イエロー・ページ

NIS (Network Information Service) を参照

一時停止した

停止させられていても、終了されていないプロセスの状態。C シェル、Korn シェル、POSIX シェルのユーザは、`fg` か `bg` コマンドを実行するか、`Ctrl-z` を押すことで、プロセスを一時停止して、再開することができます。停止されたプロセスは“一時停止されたジョブ”と呼ばれます。

ターミネートする/終了する も参照

イベント

トランスポート・ユーザにとって影響のある出現またはハプニング。イベントは非同期であり、その意味でユーザがとるアクションの結果として発生するものではありません。

イベント管理

トランスポート・プロバイダがトランスポート・ユーザに重大なイベントが発生したことを通知するメカニズム。

イベント実行可能イメージ

物理的メモリに配置されている実行可能なイメージ。

インクルード・ファイル

関数、プログラム、またはユーザのグループが使用する宣言を含むテキスト・ファイル。ヘッダ・ファイルとも呼ばれます。

`#include` も参照

インターネット

1) 主要調査機関、大学、および政府系研究所からの参加者からなるコンピュータ・ネットワークの集合であり、その中には National Science Foundation (NSF) や NFSnet 地域組織などが含まれています。インター

ネットは商用製品ではなく、どちらかというと調査用途の大規模プロジェクトであると言えます。

2) TCP/IP プロトコルを採用して接続されたネットワークの集合。

インターネット・アドレス

インターネット・ネットワークへのホストの接続を識別する固有の 32 ビットの番号。インターネット・アドレスはネットワーク番号とホスト番号からなります。

インターネット・ドメイン名システム

インターネットのドメイン名システムであり、ホストのカテゴリには COM, EDU, GOV, MIL, NET, ORG, および ARPA があります。
通信ドメイン, ドメイン名システム, インターネット も参照

インターネット・プロトコル

IP (インターネット・プロトコル) を参照

引用する

引用符で囲み、またバックスラッシュ文字 (\) の後に続けることで、ある文字がプログラムによって特別な意味で解釈されることを防ぎ、文字どおりに受け取らせること。

エスケープ も参照

インライン編集

ユーザが現在または先に入力したコマンド行を編集できるようにする一部シェルの機能。

う

ウイルス

システム内の他のコンピュータ・プログラムまたはファイルにそれ自身を添付し、すべての利用可能な手段 (ディスク・ファイル, ネットワークなど) 使用して、他のコンピュータにそれ自身を無限に複製するよう意図されたソフトウェア。ウイルスは通常「感染した」プログラムまたはシステムに損害を与えたり破壊したりするよう設計されており、しばしばウイルスの制作者の誕生日などの特定の時間に破壊的になるようにプログラムされています。

トロイの木馬, ワーム も参照

え

エスケープ

1) バックスラッシュ (\) を先頭につけることによって、プログラムによる文字の解釈を回避すること。

2) 一定のアクティビティを止めるためのコマンド、または特定の関数を実行するシーケンスの最初の文字として通常解釈される ASCII 文字。多数の端末とワークステーションのカーソル制御シーケンスがエスケープ文字を使用します。

引用する も参照

エディタ

テキスト・ファイルのコンテンツを修正するプログラム。vi のような全画面エディタはビデオ表示端末を使用して、操作するファイルの複数行を表示します。エディタに従って、ユーザはカーソルを特定の位置に移動して、そのテキストを変更します。ed のような行エディタは 1 行単位で作業を行います。sed のようなストリーム・エディタは、ユーザからコマンドを受け付けるのではなく、先に用意されているリスト (いわゆるスクリプト) からコマンドを適用して作業を行います。

エディタ

テキスト・ファイルの内容を修正するための行指向プログラム。ex エディタは ed エディタの拡張バージョンです。

エラー

操作によって期待される結果が得られない状況。XTI では、実行プロセス中にエラーがシステムに発生するか、またはライブラリ・エラーが発生すると、関数が戻されるインディケータ。オブジェクトは、戻されたエラー・コードに基づいてアプリケーションがアクションをとれるように設計されています。

演算子

正規表現で、そのリテラルな意味以外の意味を持つ文字。たとえば、1 対のカッコ ([]) はカッコによってくくられた文字のどれか 1 文字に照合するための演算子となります。

お

オープン・システム

OSI (Open System Interconnection : 開放型システム間相互接続) のための ISO (International Organization for Standardization : 国際標準化機構) 参照モデルをサポートするシステム。

大文字小文字を区別する

大文字と小文字を区別することができます。大文字小文字を区別するデバイスまたはプログラムは、A と a を異なる文字として認識します。Tru64 UNIX オペレーティング・システムの一部をなすデバイスおよびプログラムは大文字小文字を区別します。

case insensitive (大文字小文字を区別しない) も参照

オブジェクト・ファイル

コンパイラによって生成された非実行可能中間バイナリ・ファイル。オブジェクト・ファイルは、完全な実行可能バイナリのコンパイルに使用するコンパイル済みのプログラム要素を提供するために、しばしばライブラリとして使用されます。

バイナリ・ファイル, コンパイラ も参照

オプション

1) どシェルがどのようにコマンドを実行するか指定する引数。オプションは通常ハイフンの後につけられ、たとえば `ls -a` のようにコマンド行でコマンド名とともに示されます。オプションはしばしばフラグまたはスイッチとも呼ばれます。

2) スイッチの設定を示す指標またはパラメータ。

3) 単語の終わりなど特定の条件が発生したときシグナルを送る文字。

4) CPU に条件を記述する内部標識。

親ディレクトリ

別のディレクトリを含むディレクトリ。親ディレクトリに含まれるディレクトリはサブディレクトリと呼ばれます。

親プロセス

子プロセスと呼ばれるその他のプロセスを作成したプロセス。Tru64 UNIX システムでは、シェル組込みコマンドでないすべてのコマンドが子プロセスを作成します。

フォークする (*fork*) も参照

か

カーソル

ビデオ表示画面上におけるキーボード入力の位置を示すシンボル。次の文字が表示される位置にカーソルは配置されます。

ポインタ も参照

カーソル移動キー

通常は上向き，下向き，および右向き矢印がラベル付けされたキーのセットであり，それによってビデオ表示画面のカーソルが配置されます。

カーネル

ユーザとアプリケーションに基本的なサービスを提供する，オペレーティング・システムの重要な部分。カーネルは以下の機能を持っています。

- それぞれのタスクにメモリや実行時間などのシステム資源を分配する
- ユーザがモニタとキーボードを介してコンピュータとやり取りし，またタスクが大容量ストレージやその他の装置にアクセスできるように入力と出力サービスを提供する
- ネットワークの他のコンピュータとの通信を管理する
- ユーザとタスクのために認証サービスを提供する
- ファイル・システムの編成と統一性を管理する

カーネルは，それが実行されるハードウェアに固有の情報を持っています。
シェル，マイクロカーネル も参照

解析順序

プログラムが入力される情報を翻訳する順序。たとえば，左から右への解析順序を使用しているプログラムは「数列を作成；数列を書き込み」という入力を解釈して，最初のステップで作成された数が次に書き込まれるようにします。右から左の解析順序を持つプログラムは，同じ入力をまず以前のステップで作成された数列を書き込み，次に新規の数列を作成するように解釈します。

開放型システム間相互接続

OSI (開放型システム間相互接続) も参照

隠しファイル

名前がピリオドで始まるファイル。デフォルトで、ls コマンドはこのようなファイルをリスティングから省きます。

隠し文字

印刷できない ASCII 文字セットの文字。たとえば、DEL および ESC 文字。

拡張文字

7 ビットの ASCII 文字以外の文字。拡張文字を、8 ビット・セット (順序 128 ~ 255) を持つ 1 バイト・コード・ポイントにすることもできます。

仮想記憶

プロセスとファイル・システムは限られた物理メモリを取り合うため、スワップ空間またはディスク (ページング) にその内容を書き込むことで、仮想記憶サブシステムは定期的に最も古いページを再生利用します。高負荷の下では、すべてのプロセスがメモリを解放するために一時停止 (スワップアウト) されることもあります。

仮想ファイル・システム

VFS (Virtual File System : 仮想ファイル・システム) を参照

稼働時間

マシンが利用可能である時間帯。

ダウン・タイム も参照

環境

ユーザがコンピュータ上で作業している状況のセット。環境には、作業中のディレクトリの名前、コマンド・インタプリタの名前、ユーザの端末の ID などの情報が含まれます。

環境変数

シェルまたはコマンドが使用できる情報を含むシンボル。環境変数は所定のプロセス・グループ内のすべてのプロセスに使用できます。また、子プロセスを作成することによって伝播されます。

プロセス変数 も参照

完全パス名

絶対パス名 を参照

管理情報ベース

MIB (管理情報ベース) を参照

き

キーワード

- 1) 情報を検索する際に、照合される単語。
- 2) ファイルで必要とされる予約語。

疑似 **tty**

疑似ターミナル を参照

疑似ターミナル

実質的にキーボードとディスプレイ装置として動作する特別なファイル。
pty または疑似 tty と呼ばれます。

疑似デバイス も参照

疑似デバイス

疑似ターミナルなど、ハードウェアではなくソフトウェア・シミュレーションにより構成されるデバイス。

キャリッジ・リターン

後に続くテキストを次の行の左端に強制的に改行する文字、またはユーザ
入力の終わりを知らせる文字。Return キーが、キャリッジ・リターンを
生成するのに通常使用されます。

キュー

処理されるのを待つ項目の列。たとえば、プリント・キューはプリントされ
るのを待つジョブから成り立ちます。

キュー・デーモン

未処理のジョブのリストを管理し、それらのジョブを適切な時間に特定の
デバイスに送るプロセス。

キュー、デーモン、ジョブ も参照

キューに入ったメッセージ

後でユーザが表示できるように、ファイルに保存されたメッセージのリス
トに追加されるシステム・メッセージ。キューに入ったメッセージは通常
バックグラウンドプロセスで作成されます。一般的に、ユーザと直接対話
しているプログラムは、ユーザがすぐ見ることができるようスクリーン
にメッセージを送ります。

キュー要素

キューの中の項目。

行エディタ

一度に 1 行のテキストを編集する対話型，または非対話型テキスト・エディタ。

全画面エディタ も参照

共通インターネット・アドレス表記法

インターネット・ネットワーク上で，32 ビットのインターネット・アドレスに対応した 10 進法。ドット 10 進数表記法とも呼ばれます。

切離しジョブ

ユーザがログアウトしてからも処理を継続するジョブ。

く

クエリ

- 1) データについて必要な情報を検索する処理。
- 2) データ通信で，発信端末が着信端末の識別情報を要求し，適切な権限を与える過程。
- 3) 対話型システムで，システムから応答を引き出すための端末またはワークステーションでの操作。
- 4) 特定の条件に基づいてファイルから情報を求める要求。

クライアント

サーバと呼ばれる別のコンピュータ・システムが提供するリソースを使用するコンピュータ・システム。

クライアント・プロセス

通信のクライアント・サーバ・モデルで，サービスをサーバ・プロセスに要求するプロセス。

グループ

- 1) 保護リソースのアクセス権限を共有できるユーザの集合。
- 2) 単一名で一括される名前のリスト。
- 3) すべてのレコード内で特定のフィールドについて同一値を持つ関連レコードのセット。
- 4) 論理的に結合される一連のレコード。

ログイン・グループ も参照

グループ ID (GID)

関連ユーザのグループに割り当てられる固有の番号。多くの場合、グループ番号はグループ名を引数とするコマンドで代用できます。

ユーザ ID (UID), プロセス ID (PID) も参照

グローバル

プログラミング言語で、プログラムのある小区分で定義された情報であり、なおかつプログラムの少なくともそれ以外の小区分でも使用されていること。または、1 つ以上のプログラムまたはサブルーチンで使用できる情報であること。

グローバル探索

編集環境で、システムが文書全体をながめて特定の文字、ワード、または文字グループを探すプロセス。

グローバル文字

ワイルドカード文字 を参照

け

現在のディレクトリ

作業ディレクトリ を参照

こ

更新インストール

ディスク・パーティション、ファイル・システム、ファイルのカスタマイズ、ネットワーク・印刷・電子メールの環境、ユーザ・アカウント、ユーザが作成したファイル、その他のすべてのシステム設定を保持したまま行うインストール。

フル・インストレーション も参照

構成

- 1) データ処理システムまたはネットワークを構成するマシン、デバイス、およびプログラム。
- 2) オペレーティング・システムの実行によって使用可能になるサブシステムまたはサブシステムのセットを作る行為。
- 3) オペレーティング・システムに構成されたサブシステムのセット。

構成ファイル

システムまたはサブシステムの特性を特定するファイル。

構造

特定の目的で使用するデータ構造。

コール

プログラミング言語で、サブルーチン、関数、またはプロシージャのいずれかを呼び出す文。

国際化

コンピュータの機能を異なる母国語、現地の慣習、およびコード化文字セットの要件に適合させること。つまり、国際化したプログラムとは、サポートされているロケールであれば、そのプログラムを修正しなくても必ず動作することを意味します。

国際化という用語はしばしば I18N という短縮形で表現されます。これは、ワードの最初と最後の字の間に 18 字あることを示しています。

国際標準化機構

ISO (国際標準化機構) を参照

コネクション指向モード

確立された接続上でデータを送信するトランスポート終端がサポートするサービスのモード。

コネクションレス・モード

データ送信のために接続を確立する必要のないトランスポート終端がサポートするサービスのモード。データはデータグラムと呼ばれる内蔵型ユニットで伝送されます。

子プロセス

親プロセス を参照

コマンド・インタプリタ

特定のソース言語で記述されたプログラムを理解して実行するプログラム。プログラムを解釈するほうがプログラムをコンパイルする場合より実行速度が遅くなるのは、インタプリタが同時に 2 つの操作を実行しているからです。

Perl と JavaScript は、コマンド・インタプリタに依存した普及しているスクリプティング言語の一例です。UNIX シェルはコマンド・インタプリタです。

コンパイラ, シェル も参照

コマンド・インタプリタ

シェル を参照

コマンド置換

引数としてのコマンドの出力を, そのコマンド行を低アクセント (`) で囲むことによって, 別のコマンドに取り込む機能。まず, シェルがコマンドまたは低アクセントで囲まれたコマンドを実行してから, 低アクセントを含む式全体を出力に置換します。この機能は代入文で頻繁に使用されます。

コマンド・ヒストリ

ヒストリ・リスト を参照

コマンド・モード

ユーザがコマンドを入力できるシステムまたはデバイスの状態。

コメント・アウト

プログラムまたは文書ソース・ファイルの一部の解釈を選択的に無効することです。

コンパイラ

特定のソース言語で記述されたプログラムを実行可能なバイナリ・ファイル (またはオブジェクト・ファイルと称される中間バイナリ・ファイルに翻訳するプログラム。入力には, 1 つ以上のソース言語ファイルを 1 つ以上のオブジェクト・ファイルとあわせて含めることができます。プログラムをコンパイルするほうがプログラムを解釈する場合より実行速度が速いのは, コンパイラがすでに解釈を実行しているからです。cc プログラムは C コンパイラのことです。

バイナリ・ファイル, コンパイル, コマンド・インタプリタ, オブジェクト・ファイル も参照

コンパイル

1 つ以上のプログラム・ソース・ファイルを処理して, 実行可能なバイナリ・ファイルまたはオブジェクト・ファイルを生成することです。

コンパイル時間

コンパイルがプログラムのコンパイルを行っている間のアクションを指します。

実行時 も参照

コンピュータ・ウイルス
ウイルス を参照

コンピュータ・ワーム
ワーム を参照

さ

サーバ
クライアントと呼ばれる 1 つ以上のその他のコンピュータにリソースを提供するコンピュータ・システム。

サーバ・プロセス
クライアント/サーバ・モデルの通信で、クライアント・プロセスにサービスを提供するプロセス。

再帰的
プログラミングで、プロシージャまたは関数について、特定の条件が満たされるまで繰り返してそれ自身を呼び出して、その作業を遂行すること。再帰的なプロシージャまたは関数を実行するプロセスは再帰呼ばれます。

最適化
プログラムがタスクを行う際に、最も効果的に時間、I/O、その他のリソースを使用するように特定の方法を選択するプロセス。

サイレント文字
隠し文字 を参照

作業ディレクトリ

- 1) プログラムが処理するファイル名にディレクトリ・パスが含まれないときに、ファイルが読み取られる、またはファイルが書き込まれるディレクトリ。
 - 2) ユーザの現在のディレクトリ。
- ホーム・ディレクトリ も参照

サブセット

Tru64 UNIX `setld` ユーティリティとともにインストール、または削除されたソフトウェアキットモジュール。サブセットは通常、アプリケーションとそのサポートファイルのような、関連したファイルのコレクションから成り立ちます。

サブセット依存性

特定のサブセットを必要とする状態，または正常に動作するためのその他のサブセットが不足している状態。 `setld` ユーティリティの管理の下でサブセットのサブセット管理プログラム (SCP) によって判定されます。

サブセット管理プログラム

SCP (サブセット管理プログラム) を参照

サブディレクトリ

別のディレクトリに含まれた (ネストした) ディレクトリ。サブディレクトリを含むディレクトリは親ディレクトリと呼ばれます。

し

シェル

ユーザまたは別のプログラムによって入力されたコマンドを理解して，実行するオペレーティング・システムの構成要素。最も多用される UNIX シェルは **Bourne** シェル，**C** シェル，**Korn** シェル，**POSIX** シェルです。これらのすべてが Tru64 UNIX オペレーティング・システムに含まれます。

スクリプト も参照

ジオメトリ

特定のディスク装置のシリンダ，トラック，およびセクタのサイズ (バイト単位)。

ディスク・ラベル も参照

式

- 1) 値の表現方法。たとえば，変数と定数が単独で，または演算子との組み合わせを伴って表現されること。
- 2) プログラミング言語における，1 つ以上のオペランドから値を演算する言語構造。例，リテラル，識別子，配列参照，および関数呼び出しなど。
- 3) サインの構成。

字句解析プログラム

入力の解析を補助するために入力を分析し，その要素をカテゴリーごとに割り当てるプログラムやプログラムの断片。 `lex` プログラムは字句解析プログラムの作成を支援します。

パーサ も参照

字句解析ルーチン生成システム

lex を参照

システム・コール

ファイル・システムとカーネルの通信機能にアクセスする機能。

システム負荷

すべてのプロセスがコンピュータにかける負荷。システム負荷は通常、数として表され、1.0 がシステム資源の 100 パーセントの使用を表し、0.1 が 10 パーセントの使用を表します。

実行可能ファイル

コンパイラが作成したデータ・ファイルであり、そこにはコンピュータが読み込んだり、解釈したり、実行したりできるプログラム情報が含まれています。イメージまたはバイナリ・ファイルとも呼ばれます。

実行時

実行中にプログラムまたはシステムがする処理を指します。

コンパイル時間 も参照

実効ユーザ ID

現在のユーザ ID のことですが、必ずしもユーザの ID ではありません。たとえば、ログイン ID でログインしたユーザが別のユーザ ID に変更する場合があります。ユーザが変更する ID は、ユーザが元のログイン ID に戻すまでは、実効ユーザ ID になります。

実効ルート・ディレクトリ

システムがファイル検索時に開始するポイント。そのパス名はスラッシュ (/) が語頭にきます。

従属サブセット

サブセットが適切に機能するためにそれ以外のサブセットの存在を必要とする可能性のあるサブセットの状況。setld ユーティリティの制御下にあるサブセットのソフトウェア制御プログラム (SCP) によって評価されます。

SCP (サブセット管理プログラム)、サブセット も参照

従属ファイル

従属物 を参照

従属物

従属ファイル とも呼ばれます。 `make` ユーティリティで、構築されるファイル(ターゲット) が依存する要素。 ソース・ファイルはオブジェクト・モジュールの従属物です。

上位互換性

小規模のマシンでの使用のために設計されながら、より大規模のマシンの上でもに変更なしで実行できること。

条件付きコンパイル

プログラムのコンパイル中のプロセス(コード・ブロック)の一部であり、検討中のコード・ブロックに対する変数または外部条件によって有効または無効になることです。たとえば、あるプログラムに、Tru64 UNIX システム上で実行される場合のみコンパイルされるようなブロックが含まれることを言います。

条件付き実行

プログラム実行中のプログラムの動作または出力の一部であり、変数または条件によって有効または無効になることです。たとえば、あるプログラムに、プログラムをメニュー・モードで開始する場合のみユーザに問い合わせるようなコードが含まれることを言います。

条件文

プログラミング言語における文(例、`if` 文)であり、1 つ以上の変数または条件を評価してからその結果を使用して、連続するコードで複数の選択可能なパスから 1 つを選択します。

状態

プログラムが存在する状態。

ジョブ

1) ユーザによって定義され、システムによって実行される作業の単位。ジョブはまた、一連のプログラム、ファイル、制御文などオペレーティング・システムに対して行われる作業を指すこともあります。

2) 適切なジョブ制御文によって識別されて、ひとつのプロシージャにまとめられた、一連の関連するプロシージャやプログラム。

ジョブ・キュー

システムに処理されるのを待つジョブのリスト。

ジョブ状態

システムによって実行されている作業の状態。

ジョブ制御

バックグラウンド・プロセスをモニタしてアクセスするための機構。

ジョブ番号

システムに入るジョブに割り当てる番号。

所有者

通常、ファイルを作成するユーザ。所有者は、そのファイルへのアクセスを許可するユーザやグループのリストと、ユーザやグループがファイルにアクセスする方法を変更する権利を持っています。ファイルの所有権はシステム・マネージャまたはスーパーユーザによっても再設定できます。

シリアル・ライン・インターネット・プロトコル

SLIP (シリアル・ライン・インターネット・プロトコル) を参照

シンプル・メール転送プロトコル

SMTP (シンプル・メール転送プロトコル) を参照

シンボリック・リンク

別のファイルまたはディレクトリのパス名を含み、そのファイルまたはディレクトリへのポインタの役を務めるファイル。シンボリック・リンクは同じファイル・システムの中で、または別のファイル・システムの間で作成できます。ソフト・リンクとも呼ばれます。

ハード・リンク も参照

信頼できるホスト

パスワード情報を入力する必要がないアクセスを認めるネットワーク内のコンピュータ。

す

スイッチ

オプションの別名。

スーパーユーザ

通常のファイル・アクセスの制限、プロセス管理などに優先する特権を持つユーザ。 `su` コマンドを実行することで、またはシステムに `root` として

ログインすることで、これらの特権を所有するユーザはスーパーユーザになることができます。

スクリプト

コンパイルされたプログラムに対し、解釈され、指定されたシェルによって実行されるプログラム。一般に、スクリプト言語はCとC++のような高度に構造化されたコンパイラ型の言語より、容易にかつすばやくコードを書くことができます。スクリプト言語は一般にコンパイラ型の言語より処理に時間がかかりますが、短いプログラムを作成するには便利です。

ステートメント

ソース言語の命令、シェル・スクリプト、コマンド言語など。

ストライピング

ディスク入出力を分配してスループットを改善する方法。チャンクまたはストライプと呼ばれるストライプされたデータはブロックに分けられて、配列の中の複数のディスクに分配されます。ストライプすることで並列 I/O ストリームが同時に別のデバイス进行操作でき、I/O 操作を複数のデバイスで同時に処理できます。

RAID (Redundant Array of Independent Disks : 独立したディスクの冗長配列)、ミラーリング も参照

ストリーム

System V システムのために開発され、現在 UNIX システムで広く使用されている TCP/IP 定義。

ストリーム・エディタ

ユーザからコマンドを受け取る代わりに、スクリプトと呼ばれるあらかじめ準備されたリストでコマンドを設定して、テキスト・ファイルでデータを扱うプログラム。**sed** のような強力なストリーム・エディタは、完全な機能を備えた対話型ライン・エディタで利用可能なすべての操作を行うことができます。

ストリーム・ソケット

トランスポート接続経由で双方向のバイト・ストリームを提供するソケット。

スプーリング

ファイルを予約されたディスク領域にコピーし、デバイスが新規のファイルを受け取る準備ができたとき、次にアクセスしてくるデバイスに一時的なコピーを順番に配送するプロセス。一時的なコピーは作成された順序でデバイ

スに配送され、配送が完了したときに削除されます。すなわち、スプーリングは FIFO (先入れ先出し) バッファリングの一種です。スプーリングの最も一般的な使用はプリントです。プリンタが利用可能になるまでユーザを待たせる代わりに、システムは出力されるべきファイルをスプールします。そうすればユーザは元のコピーを編集または削除することができます。

せ

正規表現

どのように文字を解釈するか定義する一連の規則に従って構成され、テキスト情報を見つけるために使われる 1 つ以上の文字のパターン。たとえば、ピリオドは入力中でどの文字に対しても有効な照合文字と解釈されます。正規表現 `a.c` は文字 `a` と `c` の間に 1 つの文字を持つどのような文字列とも一致します。たとえば、`abc`、`a?c`、`a9c` などです。

パターン・マッチング も参照

制御文

プログラミング言語で、評価またはテストの結果に応じて、確実に異なるアクションを起こすことができる文。

制限付きシェル

限定された機能を持つ、管理されたシェル環境を提供するセキュリティ機能。

セキュリティ

偶然または意図的な破壊、損害、流出からデータ、システム運用、装置を保護すること。

セッション

端末セッション を参照

絶対パス名

ルート・ディレクトリで始まるパス名であり、常にスラッシュ (/) が語頭に付きます。たとえば、`/usr/games` は、絶対パス名です。完全パス名とも呼ばれます。

相対パス名 も参照

全画面エディタ

一度に画面全体を表示するエディタ。ビジュアル・エディタとも呼ばれます。行エディタ も参照

そ

相対パス名

ユーザの作業ディレクトリから始まるパス名。相対パス名が先頭のスラッシュなしで表現されます。たとえば `docs/myfile.txt` は相対パス名です。
絶対パス名 も参照

ソース階層

ソフトウェア・キットを作成する際に、`kits` コマンドにより、キットのためにサブセットにコンパイルされるディレクトリ・ツリーとファイル。
ターゲット階層 も参照

ソース・コード管理システム

SCCS (Source Code Control System) : ソース・コード管理システム) を参照

ソート

設定可能な基準に基づいてある順序でファイルの情報を整理すること。

ソケット

プロセス間通信での、通信の終端。また、ソケットと対応するデータ構造体を作成するシステム・コール。

ソケットペア

双方向通信のために UNIX ドメインで作成できるソケットのペア。パイプと同じく、ソケットペアは関連するプロセスと通信できる必要があります。
パイプ も参照

ソフト・リンク

シンボリック・リンク を参照

た

ターゲット

`make` ユーティリティで、依存部から構築される要素。実行可能プログラムは 1 つ以上のオブジェクト・モジュールからの構築されるターゲットです。
ターゲット・ファイルと呼ばれることもあります。

ターゲット階層

ソフトウェア・キットを作成するため、`kits` コマンドによってソフトウェア・キットが置かれるディレクトリ・ツリー。
ソース階層 も参照

ターミネートする/終了する

1) SCSI バスの端点で特殊な抵抗器 (ターミネータ) を設置すること。いくつかの SCSI デバイスでは手作業でターミネータを挿入/除去する必要があります。スイッチまたはソフトウェア・コマンドにより有効・無効を切り替えられる組み込み型のターミネータを持つデバイスもあります。SCSI ホスト・アダプタとそれに接続された SCSI デバイスが適切にターミネートされなければ確実に動作しないことがあります。

2) プロセスを恒久的に終了 (kill) させること。終了させられたプロセスは“終了させられたジョブと”呼ばれます。

一時停止した も参照

タームキャップ・データベース

端末の種類と機能の記述を含むファイル。特定の物理的な端末が管理される方法を判定するために `tset` コマンドと BSD Curses ライブラリ・ルーチンによって使用されます。

terminfo データベース も参照

代入文

一定のフィールドまたはパラメータの値を設定する文。プログラム・ソース・ファイルおよびスクリプトでは、代入文が `パラメータ = 値` という形式になることが多くあります。

ダウン・タイム

マシンが使用不可能である期間。

稼働時間 も参照

タスク

1) 定義された活動。たとえば、ユーザ・タスク、サーバ・タスク、プロセス・タスクなどが

2) プロセスと、プロセスを実行するプロシージャ。

単一ネットワーク管理プロトコル

SNMP (単一ネットワーク管理プロトコル) を参照

単語識別子

空白によって区切られ、シェルによって一意の要素として認知されるコマンド行。キーストロークを保存するために使用されます。単語識別子を使用することで、ユーザが現在のコマンド行で使用するために、以前のコマンド行の一部を選択できます。

探索パス

実行可能なファイルとその他の種類のファイルが検索される，通常コロンで分割された，ディレクトリの完全パス名のリスト。ユーザは `path`，`$PATH`，`MANPATH` などのような変数を定義して探索パスを作成できます。

端末セッション

ユーザがログインしてログアウトする間のコンピュータとの対話作業。

ち

チルド置換

POSIX，**Korn**，**C** シェルで，パス名の最初の文字としてチルド `()` を使用すること。デフォルトでは，シェルはユーザのホーム・ディレクトリのパス名としてチルドを解釈します。`rolf`というログイン名のユーザがパス名として `~/docs/figure_1` と入力するなら，システムは `/usr/users/rolf/docs/figure_1` とエントリを展開します。チルドをユーザのログイン名の前につけた場合，シェルはこの組み合わせを指定されたユーザのホーム・ディレクトリへの参照として解釈します。たとえば，システム上のどのユーザが入力した場合でも `~willy` は `Willy` のホーム・ディレクトリへのパスを表します。

つ

追加型バックアップ

ファイルをコピーするプロセスであり，最後のバックアップが作成されてから読み取り専用アクセス以外の理由で開かれて，また，バックアップの頻度基準に対応します。

通信ドメイン

システムのプロセス間通信機能が，ネットワークのプロパティを定義するために使用する抽象的な表現。プロパティには通信プロトコルのセット，名前を操作して解釈する規則，およびアクセス権を送信する機能が含まれます。

ツリー構造

1) ほとんどのオペレーティング・システムにおけるディスク・ディレクトリの編成。ディレクトリはファイルまたはサブディレクトリと呼ばれる別のディレクトリ，またその両方を持つことができます。さらに，サブディレクトリがそれ自身のサブディレクトリを持つことができます。結果的に，図示されたとき，この構造は木が枝分かれする様子に類似します。

2) ディスク・ディレクトリについて述べられたものと同様の方法でのデータの編成。ファイルの一般的なツリー構造はいくつかの種類があります。それぞれのデータが0か1のどちらかの2つの下位の要素(子と呼ばれる)を持つものはバイナリ・ツリーと呼ばれます。それぞれのデータが2つ以上の子を持つことができ、特定のレベルでの要素(子)が同じか、近い数を持つようにツリーの要素が平均して分配されるものはB+ ツリーと呼ばれます。

て

定義済み変数

C シェルにより定義、保守されるシェル変数。

ディスク・パーティション

パーティション を参照

ディスク・ラベル

ディスク情報のことであり、通常はセクタ0(ゼロ)に位置しており、ディスク形態および partition divisions (部分分割)が含まれます。この情報は、システム・ドライバおよびブート・プログラムがドライバを識別して、ドライバのプログラム方法やファイル・システムの探索場所を決定するのに使用します。ジオメトリ、パーティション も参照

ディレクトリ

名前を含みその他のファイルまたはその他のディレクトリの情報を制御するファイルのタイプ。

ディレクトリ階層

ファイル・システム内のディレクトリの配置。ルート・ディレクトリはディレクトリ階層の最上位であり、そこにはすべてのファイル・システムやシステム上のすべてのディレクトリへのポインタが含まれています。

ディレクトリ・スタック

後で再呼び出しをするディレクトリを格納するデータ構造。

データグラム

トランスポート・プロバイダのコネクションレス・サービスによってネットワークに伝送されるデータのユニット。ユーザ・データに加えて、データグラムには送信に必要な情報が含まれます。データグラムは内蔵型であり、先に伝送されたデータグラム、または連続して伝送されたデータグラムとは関係がありません。

データグラム・ソケット

コネクションレス・モードで伝送する個々のメッセージからなるデータグラムを提供するソケット。

データ通信

Ethernet, 電話システム, またはサテライト・リンクなどのネットワーク手段によるコンピュータ間の情報伝送。

デーモン

ユーザに透過的なシステム管理機能を実行するプロセス。デーモンはそのタスクを自動的または定期的に実行できます。たとえば, `cron` デーモンは `crontab` ファイルに列挙されたタスクを定期的に実行します。デーモンはシステムおよびアプリケーションで生成することができます。デーモンの中には手作業で開始できるものもあります。たとえば, `binlogd` コマンドによって, 指定したファイルにバイナリ・イベント・レコードのログを取るデーモンが始動します。手作業でデーモンを開始するコマンドには通常 `d` を最後に置きます。

テキスト・ファイル

`more` コマンドを使用して読み取り, `vi` などのようなプログラムを使用して編集できるファイル。テキスト・ファイルはASCII 文字セットにより構成されています。

バイナリ・ファイル も参照

デバイス特殊ファイル

プロセスがハードウェア・デバイスにアクセスするのに使用するファイル。たとえば, プリンタにはデバイス特殊ファイルからアクセスします。

ブロック型特殊ファイル も参照

デバイス・ドライバ

ディスクまたはプリンタなどの周辺機器を制御するソフトウェア。

デルタ

RCS または SCCS ファイルで, ファイルの特定バージョンを構成する変更セット。

伝送制御プロトコル

TCP (伝送制御プロトコル) を参照

と

等価クラス

照合すると等しいとみなされる文字のグルーピングまたは文字列。たとえば、多数の言語では、同一等価クラスの大文字を小文字形式と同じであるとみなしますが、一部の言語では照合に際してアクセント文字と非アクセント文字とを区別します。

同期的な実行

トランスポート・ユーザに制御を戻す前に、トランスポート・プリミティブに固有のイベントを待つことを強制する実行のモード。

統合バッファ・キャッシュ

*UBC (Unified Buffer Cache: ユニファイド・バッファ・キャッシュ)*を参照

特殊ファイル

デバイス特殊ファイル を参照

ドメイン

ドメイン名システム を参照

ドメイン名システム

インターネット全体のホスト名をまとめるためのツリー構造化システム。

通信ドメイン、インターネット・ドメイン名システム も参照

トラップ

- 1) データ通信で、プログラムではなくハードウェアによる、特定のアドレスへの条件付きジャンプ。割り込みに似ていますが、外部のイベントではなく、実行されるプログラムの直接的な処理により起動されます。
- 2) プログラミング言語で、特定の条件が満たされたとき、必要な処理を提供するサブルーチンに分岐またはジャンプするプロセス。
- 3) UNIX システムで、シェル・スクリプトでシグナルを捉えて、スクリプト内で制御をハンドラ・ルーチンに転送する特殊なステートメント。

トラップ・ハンドラ

異常な事態がプログラムの実行中に生じた時使われる、システムによって定義されたルーチン。

トランスポート・サービス

ユーザ・プロセス間のデータ転送のために、ネットワークでトランスポート層によりセッション層に与えられたサポート。提供されるサービスにはコネクション指向とコネクションレスの 2 つの種類があります。

トランスポート終端

トランスポート・ユーザがトランスポート・プロバイダによりデータを交換できる通信パス。

トランスポート・プロバイダ

ネットワークでトランスポート層サービスを提供する伝送プロトコル。

トランスポート・ユーザ

データを送るため、またはネットワークで別のプログラムがポイントからデータを受け取るために、伝送プロトコルのサービスを必要としているプログラム。

トレースポイント

プログラムの実行を中断しないで、変数の値が出力されるソース・コード・プログラムにおける特定の位置。プログラムをテストしてデバッグするために使用されます。

ブレークポイント も参照

トロイの木馬

何か有用な目的を持つように見えながら、ユーザの知らないうちに、他のファイル、プログラム、システム自体に損害を与えたり、消去したりするために設計されたコンピュータ・プログラム。トロイの木馬の例として、ゲームをプレーする間に、密かにディスク・ファイルを消去してしまうゲーム・プログラムなどが挙げられます。

ウイルス、ワーム も参照

に

二項演算子

1) 2 つの配列、データ項目、または式上で実行される演算子を表すシンボル。二項演算子のタイプには、文字、論理、数字、および関係演算子の 4 つがあります。

2) 2 つの項を持つ算術演算子。

二地点間プロトコル

PPP (2 地点間プロトコル) を参照

入力

処理されるべきベータ。

入力リダイレクション

標準入力以外の入力ソースの仕様。

ね

ネイティブ・ソフトウェア

アセンブリ言語へ、またはコンピュータの標準的なマシン語表記 (オブジェクト・ファイル) に直接コンパイルされる言語で書かれるソフトウェア。翻訳したか、解釈されたソフトウェアよりもネイティブ・ソフトウェアはより効率的で、より速く実行できます。さらに、マシンのリソースを最も有効に使用するように調整できます。

ネーム・サービス

通信で他のプロセスを識別するためにクライアント・プロセスに提供されるサービス。

ネットワーク

情報を交換し、リソースを共有する目的で接続された複数の計算機システム。

ネットワーク・ファイル・システム

NFS (ネットワーク・ファイル・システム) を参照

は

パーサ

入力を解釈し、その入力に対してどのような動作をするべきか判定するためのプログラムまたはプログラム断片。 yacc プログラムはパーサの作成を支援します。

字句解析プログラム も参照

バージョン管理システム

ファイルの修正・設定の整理と保守を援助するソフトウェア・ツール。特に、改訂版のソース・プログラム、マニュアル、データ・ファイルに対する識別・保存・ログ採取・検索を自動化します。

バージョン管理ライブラリ も参照

バージョン管理ファイル

バージョン管理システムで、元のテキストと、それに適用された一連のリビジョン (デルタ) から成り立つファイル。RCS では、このファイルは “RCS ファイル”, SCCS では “s ファイル” と呼ばれます。

バージョン管理ライブラリ

RCS または SCCS などのバージョン管理システムの下で整理され、保守されるファイルを含むディレクトリ。

パーティション

ディスクの物理的な部分。ディスクをパーティションに分けることで、さまざまなファイル・システムを持つように割り当てることができます。たとえば、ルート・ファイル・システムは通常 `a.` という名前の最初のパーティション上にあります。 `/usr` ファイル・システムは別のパーティション、一般には `g` パーティション上にあります。パーティションを使用すればディスクを柔軟に管理して使用できますが、特定のディスクで特定のファイルについてすべての利用可能空間を無制限に使用することは制限されます。

ディスク・ラベル, ジオメトリ も参照

ハード・リンク

1) `ln` コマンドが 1 つ以上の名前を 1 つのファイルに割り当てられるメカニズム。新しい名前もリンクされているファイルも両方、同一ファイル・システムになければなりません。

2) `ln` コマンドを使用するデフォルトの結果。

シンボリック・リンク も参照

バイナリ

1) 2 という数字または 2 進法。

2) コンパイル・プロセスで作成される実行可能なファイル。

3) 考えられる 2 つの状態のうち一方を想定した状況。

バイナリ・ファイル

コンパイル・プロセスで作成されるファイル。バイナリ・ファイルには ASCII 文字セットにはないコードも含まれ、使用可能な 256 バイト値すべてを利用します。バイナリ・ファイルでは、`more` のようなプログラムを使用して読み込んだり、`vi` のようなエディタを使用して編集することもできません。

テキスト・ファイル も参照

ハイパーテキスト・マークアップ言語

HTML (ハイパーテキスト・マークアップ言語) を参照

パイプ

1 つのプログラム・ディレクトリの出力をもう 1 つの入力につなぐ構成体。パイプはコマンド行で、コマンドの間に垂直バー (|) を挿入して作成します。例:

```
% nroff inputfile -ms | lpr
```

このパイプライン処理は `nroff` コマンドで入力ファイル进行处理して、`lpr` コマンドで直接プリンタに処理されたファイルを送ります。

パイプライン処理 も参照

パイプ処理

パイプライン処理 を参照

パイプライン処理

パイプにより接続された一連のコマンド。パイプで 1 つのコマンドの出力を直接もう 1 つのコマンドの入力につなぐ過程は“パイプライン処理”または“パイプ処理”と呼ばれます。

配列

データ (変数) の集合であり、そのデータは共通名で識別されて、集合内のそれぞれの位置を表す数字でお互いを区別します。区別するための数字を添字と呼びます。

パス

コマンドによって指定された実行可能ファイルがシェル組込みコマンドでなく、パス名が入力されない場合に、シェルが検索するディレクトリの順序づけられたリスト。

\$PATH, *\$path* も参照

パス名

そのファイルへのアクセスを可能にするための一連のディレクトリに連結されたファイルの名前、すなわちファイルの完全な名前。絶対パス名はルートディレクトリに始まり、先頭スラッシュで書かれます。たとえば `/usr/users/rolf/myfile.txt` のようになります。相対パス名はユーザの作業ディレクトリに始まり、先頭スラッシュなしで書かれます。たとえば `rolf/myfile.txt` のようになります。

パス名修飾子

変数修飾子 を参照

パターン・マッチング

入力情報 (通常はテキスト) を特定の一連のシンボル (通常は正規表現) と比較し、関連性を見つける過程。

正規表現 も参照

バックグラウンド・ジョブ

バックグラウンド・プロセス を参照

バックグラウンド・プロセス

通常のコマンド行入力を妨げることなく実行されるジョブ。プロセスを開始するコマンドの最後にアンバサンド (&) 文字を続けて入力すると、プロセスがバックグラウンドで実行されます。たとえば、X Window システムの時計プログラムをバックグラウンドで実行するには、`xclock &` というコマンドを入力します。その結果、時計が一方のウィンドウに呼び出されるものの、`xclock` コマンドが入力されたコマンド行は、新規コマンドを受け付けられるようになります。

フォアグラウンド・プロセス も参照

パッケージ

Tru64 UNIX オペレーティング・システム・ローダで、共通のネーム・スペースを共有するオブジェクト要素のコレクション。シンボル名はひとつのパッケージの中で一意です。シンボルはパッケージ名で区別されるため、別のパッケージからのシンボルは同一のシンボル名を持つことがあります。

パッシブ・ユーザ

XTI トランSPORT接続で、接続を開始しなかったトランSPORT・ユーザ。クライアント・プロセス、アクティブ・ユーザ、*XTI (X/Open* トランSPORT・インタフェース) も参照

ハッシング

データの品目を格納したり取り出す目的で、探索キーをアドレスに変える方式。

パッチ

問題の修正を含むファイルもしくはファイルのコレクション。ひとつのパッチで複数の問題を修正できることがあります。ソフトウェア製品のパッチのコレクションは、パッチ・キットと呼ばれることがあります。

dupatch も参照

反復

異なる日付で同一の関数を繰り返し実行することであり、多くの場合、オブジェクトは連続する近似解を使用して結果を出します。

ひ

引数カウント

コマンド・インタプリタがコマンドに渡す引数の数、またはプログラムにあるルーチンからサブルーチン、プロシージャ、関数のうちいずれか 1 つに渡す引数の数。

引数リスト

コマンド・インタプリタがコマンドに渡す実際の情報 (引数)、またはプログラムにあるルーチンからサブルーチン、プロシージャ、関数のうちいずれか 1 つに渡す実際の情報 (引数)。

ピクセル (画素)

グラフィックス・アプリケーションでディスプレイでの最も小さい要素。ビデオ画面で、ピクセルは視覚的な画像を構成する点です。ピクセルの数が通常画像の解像度を決定します。ピクセルが多ければそれだけ解像度もよくなります。

ビジュアル・エディタ

全画面エディタ を参照

ヒストリ・リスト

C シェル および Korn シェルにおけるユーザが入力した最新コマンドのリストイング。ヒストリ・リストのコマンドは、再呼び出し、修正、および再実行に使用できます。

非同期イベント

イベント を参照

非同期実行

- 1) ひとつのプロセスまたはスレッドが終了するのを待つことなく、別のプロセスまたはスレッドの実行が開始することです。
- 2) XTI では、トランスポート・ユーザに対して、待たせることなくイベントを通知する実行モードのことです。

非同期転送モード

ATM (非同期転送モード) を参照

非ブロッキング・モード

非同期実行 を参照

標準エラー

プログラムがエラー・メッセージを書き込むファイル。一般に `stderr` と呼ばれる標準エラーファイルは、デフォルトでユーザのスクリーンに割り当てられますが、ユーザが利用可能などのようなデバイスまたはファイルに再び割り当てる (リダイレクトされる) ことができる仮想ファイルです。

標準エラー出力

標準エラー を参照

標準出力

プログラムが出力データを書き込むファイル。一般に `stdout` と呼ばれる標準出力ファイルは、デフォルトでユーザのスクリーンに割り当てられますが、ユーザが利用可能などのようなデバイスまたはファイルに再び割り当てる (リダイレクトされる) ことができる仮想ファイルです。

標準出力

標準出力 を参照

標準入力

多くのプログラムが入力データまたはコマンドを受け取るファイル。一般に `stdin` と呼ばれる標準入力ファイルは、デフォルトでユーザのキーボードに割り当てられますが、ユーザが利用可能などのようなデバイスまたはファイルに再び割り当てる (リダイレクトされる) ことができる仮想ファイルです。

ふ

ファイバー・チャネル

SCSI の 3 倍もの速度でコンピュータ装置間でデータを伝送する手段を提供するテクノロジー (現在の最高速度は 10 億 bps であり、期待される速度以上に高速)。ファイバー・チャネルは光ファイバー、同軸ケーブル、およびツイスト・ペアを使用して構築され、標準のセット、とりわけ Fibre Channel Physical and Signalling 規格、つまり ANSI X3.230-1994、ならびに ISO 14165-1 によって規定されています。

ファイル記述子

UNIX システムがファイルを識別するのに使用する符号のない小さな整数。ファイル名のついたオープン・システム・コールが出されると、ファイル記述子がプロセスによって作成されます。ファイル記述子がプロセスにすでにない場合には、ファイル記述子は存在しなくなります。

ファイル・システム

物理的または論理的大容量ストレージ上のファイルの集合体およびファイル管理構造。

ファイル・ポインタ

ファイル名を含む構造を示す識別子。

ファイル・マーク

磁気テープ上にデータ・ファイルの終わりを明白にするために記述された文字の連続。

EOF (ファイルの終わり) も参照

ファイル名展開

globbing (グロビング) を参照

フィールド

- 1) レコード内の情報の基本単位。
 - 2) `awk` では、入力レコードの 1 要素。
- レコード も参照

フィールド・セパレータ

レコード内でフィールドを分離するのに使用される 1 つ以上の文字。

フィルタ

- 1) 標準入力データを読み込み、データを修正して、それを標準出力に送信するコマンド。
- 2) 指定基準に沿ったデータ、信号、または素材を分離する装置またはプログラム。

ブール

- 1) 形式が一般的な代数に類似しているものの、変数の値が真と偽という 2 つの値に制限された代数 (George Boole が命名)。ブール代数の論理は、値が 0 と 1 の数字で表されるコンピュータの 2 進論理とうまくかみ合います。

2) この用語はブール演算子とも呼ばれ、AND, OR, NOT, EXCEPT, IF, THEN, TRUE, および FALSE などが使用されます。

フォアグラウンド・ジョブ

フォアグラウンド・プロセス を参照

フォアグラウンド・プロセス

シェルが次のコマンドを受け付ける前に、完了または中断しなければならないジョブ。つまり、ワークステーションまたは端末から入力を受信するジョブのこと。

バックグラウンド・プロセス も参照

フォークする (fork)

1) 子プロセスを作成して開始するのに使用されるコマンド。

2) fork コマンドを使用した結果。

親プロセス も参照

符号拡張された

オペランドに上位ビットを付加して、コンピュータのデータ・パスより小さいオペランドのデータのサイズを大きくすること。オペランドの符号ビットが 1 なら、付加されるビットは複数の 1 です。0 なら複数の 0 のビットが付加されます。この動作はオペランドの 2 の補数の数値を保持します。

フラグ

オプション を参照

プリプロセッサ

ファイルの情報の部分を別のプログラムが理解できる形に翻訳するプログラム。たとえば、tbl プログラムは nroff テキスト・フォーマットのためのプリプロセッサです。

フル・インストレーション

Tru64 UNIX をインストールすることであり、新規のファイル・システムを作成して、オペレーティング・システムのフル・コピーをキットからシステム上にロードします。従来システム上に存在していたそれ以外のバージョンのオペレーティング・システム、レイヤード・プロダクト、およびパッチは重ね書きされます。フル・インストレーションを行うとシステム・カスタマイゼーション (例、ユーザまたはデータ・ファイル) が保持されないのは、ルート (/), /usr, および /var ファイル・システムがプロセス中に再作成されるからです。

更新インストール も参照

ブレークポイント

プログラム実行中にデバッガを停止するソース・コード・プログラムのある場所。ブレークポイントはプログラムのテストやデバッグに役立ちます。

トレースポイント も参照

プロセス ID (PID)

実行しているプロセスに割り当てられた固有の番号。

グループ ID (GID) , ユーザ ID (UID) も参照

プロセス識別

プロセス ID (PID) を参照

プロセス・テーブル

システム内のすべてのプロセスについて関連する情報を含むカーネル・データ構造。

プロセス変数

現在のプロセスのみで利用できる情報を持つシンボル。プロセス変数は子プロセスには自動的に伝えられません。

環境変数 も参照

ブロッキング・モード

同期的な実行 を参照

ブロック型デバイス

データを固定サイズのグループ内で操作するデータ記憶域または転送デバイス。たとえば、データ記憶域のサイズが通常 512 バイトのディスクなど。

文字型デバイス も参照

ブロック型デバイス・スイッチ・テーブル

Tru64 UNIX オペレーティング・システムが、特定のブロック型デバイスに関連付けられるエントリ・ポイントを選択するのに使用する方式。

文字型デバイス・スイッチ・テーブル も参照

ブロック型特殊ファイル

入力または出力デバイスへのアクセス機能を提供し、ファイル・システムをサポートできるデバイス特殊ファイル。

デバイス特殊ファイル も参照

プロファイリング

システム資源が特定のプログラムでどのように使われているかのモニタリング。プロファイリングはプログラマがプログラム・コードの効率を改善するのに役立ちます。

プロファイル・データ

プログラムがその実行時間をどのように使用しているかに関わる情報。

プロファイリング も参照

分脈探索

グローバル探索 を参照

へ

ページ

固定された物理メモリの大きさの単位。

ヘッダ・ファイル

インクルード・ファイル を参照

別名

別の名前，シンボル，またはシンボルのグループの代わりに使用する名前またはシンボル。通常は本来の表記を短縮するか，または簡単に表したものを使用します。たとえば，あるディレクトリに頻繁にアクセスする場合は，`work` というワードを “`cd /share/tomb/tools/tools/work`” の別名となるように設定できます。これによって，`work` と入力すると，`/share/tomb/tools/tools/work` ディレクトリにアクセスできるようになります。詳細については，`alias(1)` 参照ページを参照してください。

変数

プログラミング言語，シェル・スクリプト，コマンド・プロシージャなどで，その値が変化可能な記号。

変数修飾子

通常その値がパス名であるという仮定の下で，変数の一部を指す記号。

変数展開

シェルコマンド行で，その関連づけられた文字列による変数名の置き換え。

ほ

ポインタ

マウスの動きを反映して、位置を指定するシンボル。ポインタは、それが置かれる場所の機能を示すために形が変わることがあります。

カーソル も参照

ホーム・ディレクトリ

特定のユーザが所有するディレクトリであり、そこからユーザのそれ以外のディレクトリが階層上に派生しています。ログイン・ディレクトリとも呼ばれます。

作業ディレクトリ も参照

ホスト

- 1) 通信ネットワークにおける一次または制御コンピュータ。
- 2) ネットワークに組み込まれたコンピュータ。

ホスト名

ネットワーク上のコンピュータに付与された名前。

ま

マイクロカーネル

カーネルの一種で、非カーネル・コードにタスク・スケジューリング、メモリ管理、その他のサービスなどの作業の多くを任せます。マイクロカーネルは特定プロセッサのアーキテクチャ上での動作など、基本的なプリミティブだけを提供します。

マウント・ポイント

マウントされたファイル・システムの名前であるディレクトリ・ファイル。

マクロ

機械語コードにコンパイルされたとき機械語コード命令を生成する、ソース言語の一部として書かれた命令。

命令 も参照

マルチプロセッサ

2 以上のプロセッサが同一の物理的メモリを共有するシステム。

み

ミドルウェア

2つのプログラムを協調して作業させるようにするソフトウェアを指す用語。
ミドルウェアの例としては、COM for Tru64 UNIX ソフトウェアがあります。

ミラーリング

ディスク障害が生じた場合でも情報が利用可能であることを保証するため、
ディスクで情報を複製する方法。

RAID (Redundant Array of Independent Disks : 独立したディスクの冗長配
列)、ストライピング も参照

む

無限ループ

プログラムが同一の命令セットを継続的に繰り返すようなソース・コード・
エラー。たとえば、命令 A が命令 B にプログラム実行命令を送り、それ
に対して命令 B が命令 A にプログラム実行命令を送り返します。このよう
なループはプログラムの外から操作して中断するしかありません。

め

命令

コンピュータ・プログラムの一部であり、コンピュータにその段階でど
の関数を実行するかを伝えます。

メールボックス

新規と未読のメールメッセージを含むファイル。メールボックス・ファイル
は通常 `/usr/spool/mail` ディレクトリにあります。

メタ文字

コンピュータ・システムにとって特殊な意味を持たされた文字。たと
えば、アスタリスクはワイルドカード照合を行うためファイル名中でしば
しば使用されます。

メモリ巡視機能

メモリ・エラーを事前に発見して処理するために、システム・メモリを
読み込むプロセス。

メモリ・ファイル・システム

MFS (メモリ・ファイル・システム) を参照

も

モード

ファイルに設定する一連のアクセス権。これらのアクセス権はしばしば、それぞれのビットにアクセス権コードに設定する 8 進数記法の 3 つの数で表現されます。アクセスを許可または拒否されるユーザには“所有者”，“グループ”，“その他”があります。与えられる最も一般的なアクセス権は読み取り，書き込み，実行権です。アクセス権を変更するために使用するコマンドは `chmod` です。

文字型デバイス

1 文字単位でデータを操作するデータ・ストレージまたは転送デバイス。
例，端末など。

ブロック型デバイス も参照

文字型デバイス・スイッチ・テーブル

Tru64 UNIX オペレーティング・システムが，特定の文字型デバイスに関連付けられたエントリ・ポイントを選択するために使用する方式。

ブロック型デバイス・スイッチ・テーブル も参照

文字型特殊ファイル

処理が文字ストリーム指向の I/O インタフェースまたは 非構造化 (raw) デバイスのいずれかにアクセスできるファイル。例，通信回線または非バッファ型磁気テープまたはディスクなど。

ゆ

ユーザ ID (UID)

個々のログイン名と関連づけられた数値。この数値は `/etc/passwd` ファイルに保存されます。

グループ ID (GID)，プロセス ID (PID) も参照

ユーザ・データグラム・プロトコル (UDP)

UDP (User Datagram Protocol) : ユーザ・データグラム・プロトコル) を参照

ユーザ名

ログイン名 を参照

優先データ

緊急とみなされるデータ。このデータの意味規則はトランスポート・プロバイダによって定義されます。

ら

ラベル

ディスク・ラベル を参照

り

リテラル

- 1) 定数を表す値表現。
- 2) プログラムの変換中に変更できない特定のシンボル。

リビジョン管理システム

RCS (リビジョン管理システム) を参照

リファレンス・ページ

コマンド、システム・コール、ライブラリ・ルーチンなどのマニュアルを持っているファイルのコレクションの1つ。リファレンス・ページはしばしば *manual pages* を省略して *man pages* と呼ばれます。

リンキング・ローダ

実行可能なファイルを生成するために、コンパイルまたはアセンブルされたプログラム、ルーチン、サブルーチンをロード、移動、リンクするためのプログラム。リンク・ローダ、またはリンカ・ローダと呼ばれることもあります。

リンク

ファイルを参照しているディレクトリ・エントリ。

ハード・リンク、シンボリック・リンク も参照

リント

バグ、ポータビリティ問題、照合しない引数タイプ、イニシャライズされていない変数などのエラーがないかCのコードを調べるプログラム。

る

ルーティング・デーモン

ルーティング管理サービスを提供するプログラム。ルーティング・デーモン `routed` は、ネットワーク・ルーティング・テーブルを管理するためにシステムが起動する際に呼び出されます。

デーモン も参照

ループ

- 1) 特定の条件が満たされるまで繰り返し実行される一連の命令。
- 2) Tru64 UNIX 仮想メモリ・システムで、置き換えのために繰り返し走査されるメイン・メモリ内のページ・クラスタ。

無限ループ も参照

れ

レイヤード・プロダクト

Tru64 UNIX システムの機能を追加するために後でインストールできるように設計されたオプションのソフトウェア製品。

レコード

- 1) 1 つの単位として扱われる関連したデータ項目のコレクション。レコードは 1 つ以上のフィールドを含みます。
- 2) `awk` で、2 つの連続したレコード区切り文字の間の情報。ほとんどの場合、`awk` でのレコードは入力ファイルの行であると考えることができます。

連結する

並べて配置することです。文字列のようなデータを連結する場合には、最初のオリジナル文字列の文字に続けて 2 番目以降のオリジナル文字列の文字すべてを含む文字列を生成することができます。ファイルについては、同様の方法でコンテンツを新規ファイルまたはオリジナル・ファイルに組み合わせることによって結合できます。

ろ

ローカリゼーション

コンピュータ・システムで地域的な要件を実装する過程。これらの要件のいくつかはロケールで扱われます。それぞれのロケールは現地の言語、文化に依存するデータ、コード・セットの特定の組み合わせをサポートするための

一連のデータを持っています。 ロケールが持つことができる情報の種類とロケールを実行するインタフェースは標準化の適用を受けます。

“ローカリゼーション (localization)”は、最初の文字 “L” と最後の文字 “N” の間に文字が 10 あるという意味で L10N と省略されることがあります。

国際化 も参照

ローカル・エリア・ネットワーク

LAN (ローカル・エリア・ネットワーク) を参照

ローカル・エリアの転送

LAT (ローカル・エリア転送) を参照

ローカル・ホスト

ユーザ端末に直接接続されたコンピュータ・システム。

ローリング・アップグレード

ソフトウェアによるクラスタのアップグレードをクラスタの動作中に行うことです。一度に 1 つのメンバが「“ロール”」して動作状態に復帰する一方で、基本オペレーティング・システム、クラスタ・ソフトウェア、およびワールドワイド言語サポート (WLS) ソフトウェアに対して、クラスタは複数のバージョンからなる環境を自動的に維持します。サービスにアクセスしているクライアントには、ローリング・アップグレードが進行中であることはわかりません

Tru64 UNIX Version 5.0A 以上のクラスタ化されたシステムでは、Tru64 UNIX オペレーティング・システムまたは TruCluster Server ソフトウェアの更新を行う場合、またはシステムのパッチを行う場合に、ローリング・アップグレードを実行します。

ログアウト、ログオフ

ユーザが現在のセッションを終了することをオペレーティング・システムに知らせるコマンドを入力して、コンピュータ・システムの実行を中止すること。

ログイン

通常はログイン名とパスワードを入力することで、正規ユーザとしてオペレーティング・システムにアクセスし、コンピュータ・システムを使用し始めること。

ログイン・グループ

ユーザにより作成されたファイルのアクセス権を確立する主要な分類。

グループ も参照

ログイン・シェル

ユーザがシステムにログインしたときデフォルトで実行されるシェル。
passwd ファイル中のユーザのエントリで指定されます。

ログイン・ディレクトリ

ホーム・ディレクトリ を参照

ログイン名

コンピュータ・システム，そしてシステムのその他のユーザがユーザを識別するための名前。システムにログインする際に，ユーザはログイン名と（通常は）パスワードを入力します。ユーザ名と呼ばれることもあります。

ロック・ファイル

あるファイルへの操作を限定または禁止することを示すファイル。ロック・ファイル自体を指標として使用するか，ロック・ファイルに制限の内容を記述することができます。たとえば，Tru64 UNIX setld ユーティリティは，インストールするそれぞれの製品キットのサブセットのためのロック・ファイルを生成します。もし特定の製品がインストール済みの別のサブセットを必要とするサブセットを含む場合，以前のサブセットを削除してしまわないように setld が後のサブセットの名前を以前のサブセットのロック・ファイルに加えます。

ロケール

中国やフランスなど特定の地域の文化的な慣習と，その地域で使用される言語を正確に処理するためのソフトウェア環境。たとえば中国語のロケールを選択することで，すべてのコマンド，システム・メッセージ，キーストロークは漢字に対応し，中国語に適した方法で表示されます。

国際化 も参照

論理ユニット番号

LUN (論理ユニット番号) を参照

わ

ワーム

それ自身を複製し，他のコンピュータ中に感染してネットワークで広まるように設計されたコンピュータ・プログラム。ワームは他のプログラムまたはファイルにそれ自身を添付しません。ワームは通常「感染した」プログラム

またはシステムに損害を与えたり破壊したりするよう設計されており、しばしばワームの制作者の誕生日などの特定の時間に破壊的になるようにプログラムされています。いくつかのワームは直接的な損害を与えるわけではありませんが、正当な使用のために意図された資源を占拠するという点で有害です。
トロイの木馬、ウイルス も参照

固定メモリ

通常、ブート時でオペレーティング・システムに割り当てられ、ページングのためには使用できない物理メモリ。固定メモリは静的か、動的かのどちらかです。

- 静的な固定メモリはブート時に割り当てられ、オペレーティング・システム・データとテキスト、システム・テーブルのために使われます。また、最近アクセスされた UNIX File System (UFS) と CD-ROM ファイル・システム (CDFS) メタデータを保持するメタデータ・バッファ・キャッシュによっても使用されます。
- 動的な固定メモリはブート時に割り当てられ、システム・ハッシュ・テーブルのような動的に割り当てられるデータ構造のために使われます。

ワイルドカード文字

ファイル名または正規表現でワイルドカード照合を行うために使用されるメタ文字。

メタ文字、正規表現 も参照

割り込み

1) あるイベントが原因となってコンピュータが通常の処理ストリームからそれて、それをトリガした状況に対処すること。事象の誘発原因が終了すると、通常の処理ストリームが割り込み時点から再開されます。割り込みはソフトウェアの命令または I/O 操作の終了のようなハードウェア・イベントのいずれかによって引き起こされます。

2) 割り込みをトリガすること。

割り込みハンドラ

割り込みに対応するアクションを実行するプログラムまたはオペレーティング・システムのコード。

A

Advanced Server for UNIX
(ASU を参照)
Advanced File System
(AdvFS を参照)
Advanced Printing Software 1-14
AdvFS 4-3
AlphaServer GS システム 1-7
AltaVista
Tru64 UNIX に関する情報の検索に
使用 1-16
ARMTech 1-14
ARP 3-21
ASU 10-1
Web によるドキュメントの参
照 1-17
ATM 3-23
ATOM 6-4
authck ユーティリティ 7-6

B

Berkeley Internet Name Domain
(BIND を参照)
Best Practices ドキュメント
Web による参照 1-17
BGP 3-5

BIND 3-41
BOOTP 3-36
bootpd デーモン 3-36

C

Capacity on Demand 1-5
CDE 9-1
概要 9-1
**CDE アプリケーション・マネー
ジャ** 2-23
CDF 2-3
CDFS 4-10
CD-ROM ファイル・システム
(CDFS を参照)
CFS 4-4
Common Desktop Environment
(CDE を参照)
Compaq Analyze 1-11
HP Insight Manager 2-26
CPU ホット・スワップ
(OLAR を参照)
CSLIP 3-25
CSP
(パッチ・キット を参照)
Curses ライブラリ 8-12
C コンパイラ 6-2

D

DAC..... 7-3
dbx デバッグ 5-2, 6-3
DDR 2-33
DECevent..... 1-12, 2-30
dense コード・ロケール 8-8
DHCP 3-35
Division of Privileges..... 1-13
DLPI..... 3-32
DMS 2-34
DNS..... 3-4
DNS..... 3-41
dupatch..... 2-9
DVD ファイル・システム 4-10
Dynamic Host Configuration Protocol
(DHCP を参照)

E

EGP..... 3-4
envonfig ユーティリティ 1-12
ERP
(パッチ・キット を参照)
eSNMP..... 3-32
Ethernet 3-24
(Fast Ethernet; Gigabit Ethernet も参照)
Ethernet パケット・フィルタ . 3-34
Event Manager 2-29
EVM 2-29

F

Fast Ethernet 3-24

FDDI 3-24
FDFS 4-13
FFM 4-12
File-on-File マウント・ファイル・システム
(FFM を参照)
file コマンド 8-13
Finger プロトコル
(フィンガ・プロトコル (FINGER) を参照)
FTP 3-9
fverify ユーティリティ 7-5

G

gated デーモン 3-37
Gigabit Ethernet 3-24
gprof..... 6-5

H

hiprof プロファイリング・ツール..... 6-5
HP Insight Manager 1-12
HTML ファイルのバージョン
Tru64 UNIX ドキュメントの参照 1-16
hwmgr ユーティリティ . 1-12, 2-32

I

ICMP 3-22
ICMPv6 3-18
IMAP 3-12
Insight Manager
(HP Insight Manager を参照)

Internet Express for Tru64 UNIX

Web によるドキュメントの参
照 1-17
概要 1-21

I/O デバイスのホットスワップ 5-12

IP

(インターネット・プロトコル
を参照)

IPv6

コマンドおよびデーモン 3-19

ISO 10646 標準 8-3

J

Java 開発キット 6-8

JDBC

(ODBC と JDBC を参照)

K

kdbx デバッグ 5-2

L

Ladefug デバッグ 5-2, 6-3

LAT 3-40

Logical Storage Manager

(LSM を参照)

LSM 2-16

M

malloc 関数 5-11

MFS 4-11

MLD 3-17

mmap 6-10

Mobile IPv6 3-18

Monitoring Performance History

ユーティリティ
(MPH, システムの監視 を参照)

Motif 9-10

Motif

国際化 8-14

MPH 2-27

Multicast Listener Discovery

(MLD を参照)

N

Neighbor Discovery プロトコ

ル 3-17

NetRAIN 3-27

Network Setup ウィザード ... 1-10

NFS 3-9, 4-5

PC-NFS 4-9

Version 3 の機能 4-6

WebNFS 4-9

に対する拡張 4-7

NIC 3-26

niffconfig コマンド 3-40

NIS 3-42

NTP 3-43

NUMA

概要 5-5

O

ODBC と JDBC 10-2

OLAR 1-7

Open Source Internet Solutions

(Internet Express for Tru64
UNIX を参照)

OSIS

(Internet Express for Tru64
UNIX を参照)

OSPF 3-7

P

PC-NFS 4-9

pcnfsd デモン 4-9

PC コード・ページ 8-7
(コードセット変換 も参照)

PDF ファイルのバージョン

Tru64 UNIX ドキュメントの参
照 1-16

PFS 4-11

pixie 6-5

Point-to-Point Protocol

(PPP を参照)

POP 3-11

Post Office Protocol

(POP を参照)

PPP 3-25

/proc ファイル・システム

(PFS を参照)

prof 6-5

Q

Quick Setup ユーティリティ .. 2-11

R

RFC

サポート・リスト A-1

RIP 3-6

RSVP 3-12

S

Secure Web Server 1-15

setld ユーティリティ

アップデート・インストレーション

後の使用 2-5

説明 2-8

ShareExpress

(ARMTech を参照)

slattach オプション 3-25

SLIP 3-21, 3-25

SMP 5-3

SMP 1-2, 1-23

SMTP 3-11

SNMP 3-11

SNMP エージェント 3-37

SPD

(QuickSpecs を参照)

spike 1-13

STREAMS 3-31

STREAMS フレームワーク

XTI との関係 3-29

svc.conf ファイル 3-41

sys_check 2-27

SysMan Menu 2-19

SysMan Station 2-22

SysMan アプリケーション 2-18

System V 1-15

T

TCP 3-2, 3-9, 3-14

TELNET 3-10
TFTP 3-10
third degree プロファイリング・
 ツール 6-5
TIS (Thread Independent
 Services) 6-10
Tru64 UNIX のインストール ... 2-1
 アップデート・インストレーショ
 ン 2-3
 クローン・インストレーション 2-6
 フル・インストレーション.... 2-2
Tru64 UNIX の沿革..... 1-2
TruCluster Server ソフトウェア
 Web によるドキュメントの参
 照 1-17
 概要 1-18
TSP 3-44
tty サブシステム 8-12

U

UBC 5-6
 LRU ページ・リスト..... 5-7
UCS-4 エンコーディング・フォーマッ
 ト..... 8-9
 (Unicode も参照)
UCS-4 エンコーディング・フォーマッ
 ト..... 8-3
UDP 3-2, 3-9, 3-13
UFS..... 4-4
UIL コンパイラ 8-14
Unicode 8-9
Unicode ロケール 8-8
Unified Buffer Cache

(UBC を参照)

UNIX 間コピー・プログラム

(UUCP を参照)

UNIX ファイル・システム

(UFS を参照)

UTF-8 エンコーディング・フォーマッ
 ト..... 8-9

(Unicode も参照)

UTF-8 エンコーディング・フォーマッ
 ト..... 8-3

UUCP..... 3-39

V

VFS 4-2
VLAN..... 3-28
VM サブシステム 5-6
 スワップ・バッファ..... 5-12
 スワップ領域割り当てモード 5-11
 変更されたページのプリライト 5-8
 ページングとスワッピングの制
 御 5-8
 ユニファイド・バッファ・キャッ
 シュ 5-6

W

wchar_t data type 8-3
Web-Based Enterprise Services
 (WEBES を参照)
WEBES..... 2-26
WebNFS..... 4-9
Web サーバ
 (Secure Web Server を参照)

Windows

Tru64 UNIX との相互運用性 . 10-1

X

X11R6.3

国際化 8-14

X11 シェアード・ライブラリ 6-6

XDM-AUTHORIZATION-1

ディスプレイ・マネージャ 9-8

xmodmap キーマップ・フォーマット

ディスプレイ・マネージャ 9-8

X/Open トランスポート・インタ

フェース

(XTI を参照)

XTI

STREAMS およびソケット・フレ

ームワークとの関係 3-29

STREAMS ベースのトランスポー

ト・プロバイダでのデータ・フ

ロー 3-30

定義 3-29

X Window System 9-3

X クライアント・ライブラリ 9-3

X サーバ 9-4

X サーバ拡張機能 9-4

X ベース・システム管理ユーティリ

ティ 2-28

あ

アクティブ・ページ・リスト 5-7

アダプタ

多重アダプタのサポート 3-26,
3-31

圧縮シリアル・ライン **IP**

(CSLIP を参照)

アドレス

マッピング 3-21

アドレス解決プロトコル

(ARP を参照)

アプリケーション・プログラミング・

インタフェース 6-11

DLI 6-11

DLPI 6-11

STREAMS 6-11

XTI 6-11

ソケット 6-11

ネットワーク・プログラミング環境

の概要 3-28

アプリケーション・レベルのプロトコ

ル 3-3

い

イベント管理 2-26

インターネット

(TCP を参照)

インターネット制御メッセージ・プ

ロトコル

(ICMP を参照)

インターネット・プロトコル

IPv4 3-15

IPv4 マルチキャストイング .. 3-16

IPv6 3-17

インターネット・プロトコル・パー

ジョン **6**

(インターネット・プロトコル

を参照)

インターネット・ブート・プロトコ

ル・デーモン

(BOOTP を参照)

う

- ウィジェット
(拡張ウィジェット・セットを
参照)
- ウィンドウ環境
(CDE を参照)

え

- エラー処理 3-22

お

- オンラインでの追加と取り外し
(OLAR を参照)

か

- 開発環境 6-1
- 外部ゲートウェイ・プロトコル
(EGP を参照)
- 拡張ウィジェット・セット 9-10
- カスタム・セットアップ ユーティリ
ティ 2-14
- 仮想ファイル・システム
(VFS を参照)
- 仮想メモリ・サブシステム
(VM サブシステム を参照)
- 簡易ファイル転送プロトコル
(TFTP を参照)
- 環境モニタリング 1-12, 2-29
- カーネル・チューナ 1-13
- カーネル・デバッグ 5-1

- I/O デバイスのホットスワップのサ
ポート 5-12
- チューニング 5-1

き

- 境界ゲートウェイ・プロトコル
(BGP を参照)

く

- クイックスタート 6-6
- クラス・スケジューラ 1-13
- クラスタ・ファイル・システム
(CFS を参照)
- クラッシュ・ダンプ
管理 2-31
- クローニング
(Tru64 UNIX のインストール
を参照)

け

- ゲートウェイ 3-4

こ

- 構成記述ファイル
(CDF を参照)
- 構成チェックリスト 2-11
- 国際化 8-1
- Curses ライブラリ 8-12
- file コマンド 8-13
- PC コードページ・フォーマッ
ト 8-7

Unicode support 8-9
グラフィカル・アプリケーション 8-14
コードセット変換 8-7
サポート言語 8-3
端末サブシステム 8-12
プリント・サブシステム 8-13
ロケールの作成 8-7
国際化機能
システム管理 8-9
国際化ソフトウェア構成ユーティリティ 8-9
固定ページ・リスト 5-7
コネクションレス型メッセージ 3-15
コードセット変換 8-7
コードの最適化
(spike を参照)

さ

サーバ
(Secure Web Server を参照)

し

シェアード・ライブラリ 6-6
完全なあるいは部分的な重複のサポート 6-7
クイックスタート 6-6
時刻同期化プロトコル
(TSP を参照)
システムおよびネットワーク管理ユーティリティ 1-9
システム管理 2-1
グラフィカル・ユーザ・インタフェース 2-18

ハードウェアの管理 2-32
システム管理ユーティリティ
(SysMan Menu を参照)
システム構成
dynamic 2-34
システムの監視 2-26
環境モニタリング 2-29
システムの性能 2-26
実行時ライブラリ 6-7
シリアル・ライン・インターネット・プロトコル
(SLIP を参照)
シンプル・ネットワーク管理プロトコル
(SNMP を参照)
シンプル・メール転送プロトコル
(SMTP を参照)

す

スケーラビリティ機能 1-5
ストレージ管理
(LSM を参照)
スレッド 6-9
ビジュアル 6-9
スワッピング
(VM サブシステム を参照)
スワップ領域の割り当て
延期モード 5-11
即時モード 5-11
レイジー・モード 5-11

せ

性能管理 2-26
性能履歴

監視 2-27
セキュリティ 7-1
 authck ユーティリティ 7-6
 fverify ユーティリティ 7-5
 オブジェクトの再使用 7-4
 監査 7-3
 管理 7-2
 システム・アーキテクチャ... 7-4
 任意アクセス制御 7-3
セキュリティ統合アーキテクチャ層
 (SIA 層 を参照)
セットアップ・ユーティリティ . 2-11

そ

ソケット 3-31
 STREAMS の相互作用 3-32
ソケット・フレームワーク
 XTI との関係 3-29, 4-2
ソフトウェア仕様書
 (QuickSpecs を参照)

た

対称型マルチプロセッシング
 (SMP を参照)
ダイナミック・ローダ
 シェアード・ライブラリ 6-6
タイム・サービス
 NTP 3-43
 TSP 3-44
多重アダプタのサポート 3-26, 3-31
多重化 3-15
ダンプ

(クラッシュ・ダンプ を参照)
端末ドライバ・サブシステム... 8-12

つ

通信ブリッジ
 DLPI STREAMS 擬似ドライ
 バ 3-32
 ifnet STREAMS モジュール.. 3-32

て

ディスプレイ・マネージャ 9-8
デジタル汎用ディスク・ファイル・
 システム
 (DVD ファイル・システム を
 参照)
デバイス・ドライバ・ドキュメント
 Web によるドキュメントの参
 照 1-17
デバッグのためのツール 6-3
 (Ladebug デバッガ; dbx デバッ
 ガ も参照)
伝送制御プロトコル
 (TCP を参照)
データ・アクセス
 (ODBC と JDBC を参照)
データ・フロー
 XTI および STREAMS ベースのト
 ランспорт・プロバイダ 3-30
 XTI およびソケット・ベースのトラ
 ンспорт・プロバイダ... 3-30
データ・リンク・インタフェース
 (DLI を参照)

データ・リンク・プロバイダ・インタ
フェース
(DLPI を参照)
データレス管理サービス
(DMS を参照)
デーモン
bootpd 3-36
gated 3-37

と

ドキュメント 1-17
Tru64 UNIX の概要 1-16
Web 上での検索 1-17
Web 上で利用可能 1-17
オンライン・ヘルプ 1-18
ドキュメント・キット内の印刷された
ドキュメント 1-17
ドメイン・ネーム・システム・プロ
トコル
(DNS を参照)
トランスポート・レベルのプロトコ
ル 3-13
トークン・リング 3-25

に

任意アクセス制御
(DAC を参照)

ね

ネットワーク・アダプタ 3-23
ネットワーク・インタフェース
監視 3-40

ネットワーク・インタフェース・カー
ド
(NIC を参照)
ネットワーク管理 3-33
コマンドおよびユーティリ
ティ 3-33
ネットワーク情報サービス
(NIS を参照)
ネットワーク・タイム・プロトコル
(NTP を参照)
ネットワーク・ファイル・システム
(NFS を参照)
ネットワーク・プログラミング環
境 6-11
アプリケーション・プログラミン
グ・インタフェース 6-11
コンポーネント 6-11
通信ブリッジ 6-11
データ・リンク・インタフェー
ス 6-11
ネットワーク・レベルのプロトコ
ル 3-15
ネーミング・サービス 3-41
ネーミング・サービス構成ファイル
(svc.conf ファイル を参照)

は

パッケージング 1-22
バージョニング
(シェアード・ライブラリ を参
照)
ハードウェア管理
(hwmgr ユーティリティ を参照)

ひ

- 非アクティブ・ページ・リスト.. 5-7
- 標準
- Tru64 UNIX の概要 1-3

ふ

- ファイル記述子ファイル・システム
(FDFS を参照)
- ファイル・システム 4-1
 - AdvFS 4-3
 - CDFS 4-10
 - CFS 4-4
 - DVD 4-10
 - FDFS 4-13
 - FFM 4-12
 - NFS 4-5
 - /proc ファイル・システム 4-11
 - UFS 4-4
 - VFS 4-2
 - 概要 4-1
 - メモリ・ファイル・システム 4-11
- ファームウェア 1-22
- フィンガ・プロトコル
(**FINGER**) 3-10
- フォント・サーバ 9-9
- フォント・レンダ 9-9
- 不均等メモリ・アクセス
(NUMA を参照)
- プリント機能
- 国際化 8-13
- フリーウェア
- CD-ROM 上 1-16

- フリー・ページ・リスト 5-7
- 定義 5-7
- プロセス・チューナ 1-13
- プロファイリング・ツール 6-4
- 分散システム・サービス
- タイム・サービス 3-42
- ネーミング・サービス 3-41
- 分散ネーミング・サービス 3-41
(BIND; NIS も参照)
- ブータブル・テープ 1-13

へ

- ページング
(VM サブシステム を参照)

ま

- マッピング・テーブル 3-21
- マニュアル
- メディア・キットに含まれる印刷
版 1-23
- マルチキャストینگ
(インターネット・プロトコル
を参照)
- マルチヘッド・グラフィック・サポー
ト 9-4

め

- メディア・キット
(パッケージング を参照)
- メモリ・ファイル・システム
(MFS を参照)

メモリ・マップ・ファイル・サポート
（ mmap を参照 ）
メール
電子 3-39

ゆ

ユーザ・データグラム・プロトコル
（ UDP を参照 ）
ユーロ通貨記号 8-10

ら

ライセンス情報 1-23
ライブラリ 6-6
（ シェアード・ライブラリ も参
照 ）
実行時 6-7

り

リアルタイム
環境 6-10
リソース管理ソフトウェア
（ ARMTech を参照 ）

リソース・リザベーション・プロト
コル
（ RSVP を参照 ）
リンク・アグリゲーション 3-26

る

ルーティング情報プロトコル
（ RIP を参照 ）
ルーティング・プロトコル 3-4

れ

レイヤード・プロダクト
アップデート・インストレーション
中の解析 2-4

ろ

ロケール
作成 8-7
ローカル・エリア転送
（ LAT を参照 ）
ローリング・アップグレード 2-8

Tru64 UNIX ドキュメントの購入方法

Tru64 UNIX ドキュメントのご購入については、弊社担当営業または日本ヒューレット・パッカートの各営業所/代理店にお問い合わせください。

各ドキュメント・キットの注文番号は以下のとおりです。ドキュメント・キットに含まれるマニュアルの内容については『ドキュメント概要』を参照してください。

キット名	注文番号
Tru64 UNIX Documentation CD-ROM	QA-6ADAA-G8
Tru64 UNIX Documentation Kit	QA-6ADAA-GZ
End User Documentation Kit	QA-6ADAB-GZ
- Startup Documentation Kit	QA-6ADAC-GZ
- General User Documentation Kit	QA-6ADAD-GZ
- System and Network Management Documentation Kit	QA-6ADAE-GZ
Developer's Documentation Kit	QA-6ADAF-GZ
Reference Pages Documentation Kit	QA-6ADAG-GZ
TruCluster Server Documentation Kit	QA-6BRAA-GZ
Tru64 UNIX 日本語ドキュメント・キット	QA-6ADJB-GZ
スタートアップ・ドキュメント・キット	QA-6ADJC-GZ
一般ユーザ・ドキュメント・キット	QA-6ADJD-GZ
システム/ネットワーク管理ドキュメント・キット	QA-6ADJE-GZ
プログラミング・ドキュメント・キット	QA-6ADJF-GZ
CDE 翻訳ドキュメント・キット	QA-6ADJG-GZ
TruCluster Server 日本語ドキュメント・キット	QA-05SJA-GZ
Advanced Server for UNIX 日本語ドキュメント・キット	QA-5U2JA-GZ



マニュアルに対するご意見

Tru64 UNIX

Tru64 UNIX 概要

AA-RK3KE-TE

弊社のマニュアルに関して、ご意見、ご要望、または内容の不明確な部分など、お気づきの点がございましたら、下記にご記入の上、弊社社員にお渡しくださるようお願い申し上げます。

マニュアルの採点：

	大変良い	良い	普通	良くない
正確さ(説明どおりに動作するか)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
情報量(十分か)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
分かり易さ	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
マニュアルの構成	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
図(役立つか)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
例(役立つか)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
索引(項目の検索性)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
ページ・レイアウト(情報の検索性)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

内容の不明確な部分がありましたら、以下にご記入ください：

ペー ジ

その他お気づきの点がございましたら、以下にご記入ください：

ご使用のソフトウェアのバージョン： _____

貴社名/部課名 _____

御名前 _____

記入日 _____

(注) 当用紙を受け取った弊社社員は、すみやかに下記にお送りください。

ビジネスクリティカルシステム統括本部 **BCS** 技術本部 **Alpha** ソフトウェア技術部