



SunOS リファレンスマニュアル (1M) : 保守コマンド

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

Part No: 817-5306-10
2004 年 2 月

Copyright 2004 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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

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

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

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

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

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

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

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

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

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

「ATOK」は、株式会社ジャストシステムの登録商標です。

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

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

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

Unicode は、Unicode, Inc. の商標です。

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

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

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

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

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

原典: SunOS Reference Manual (1M) (man pages section 1M: System Administration Commands)

Part No: 816-3319-10

Revision A



031209@7518



目次

はじめに 5

SunOS リファレンスマニュアル (1M) : 保守コマンド 9

Intro(1M) 10
accept(1M) 13
acct(1M) 15
acctdisk(1M) 18
acctdusg(1M) 21
accton(1M) 24
acctwtmp(1M) 27
afbconfig(1M) 30
catman(1M) 39
cfgadm(1M) 43
cfgadm_ac(1M) 54
cfgadm_sysctrl(1M) 58
closewtmp(1M) 63
cvcd(1M) 66
dd(1M) 67
df(1M) 73
du(1M) 77
ffbconfig(1M) 79
GFXconfig(1M) 88
installer(1M) 93
lpmove(1M) 94
lpsched(1M) 96
lpshut(1M) 98

luxadm(1M) 99
m64config(1M) 115
modinfo(1M) 120
modload(1M) 121
modunload(1M) 123
patchadd(1M) 124
patchrm(1M) 134
pgxconfig(1M) 140
pmconfig(1M) 145
powerd(1M) 147
prodreg(1M) 148
prtdiag(1M) 149
raidctl(1M) 150
reject(1M) 153
restricted_shell(1M) 155
rsh(1M) 157
scadm(1M) 159
su(1M) 166
SUNWafb_config(1M) 169
SUNWffb_config(1M) 178
SUNWm64_config(1M) 187
ttypmon(1M) 192
utmp2wtmp(1M) 196
wall(1M) 199

はじめに

概要

SunOS リファレンスマニュアルは、初めて SunOS を使用するユーザーやすでにある程度の知識を持っているユーザーのどちらでも対応できるように解説されています。このマニュアルを構成するマニュアルページは一般に参照マニュアルとして作られており、チュートリアルな要素は含んでいません。それぞれのコマンドを実行すると、どのような結果が得られるかについて、詳しく説明されています。なお、各マニュアルページの内容はオンラインでも参照することができます。

このマニュアルは、マニュアルページの内容によっていくつかのセクションに分かれています。各セクションについて以下に簡単に説明します。

- セクション 1 は、オペレーティングシステムで使えるコマンドを説明します。
- セクション 1M は、システム保守や管理用として主に使われるコマンドを説明します。
- セクション 2 は、すべてのシステムコールについて説明します。ほとんどのシステムコールに 1 つまたは複数のエラーがあります。エラーの場合、通常ありえない戻り値が返されます。
- セクション 3 は、さまざまなライブラリ中の関数について説明します。ただし、UNIX システムプリミティブを直接呼び出す関数については、セクション 2 で説明しています。
- セクション 5 は、文字セットテーブルなど他のセクションには該当しないものについて説明します。

以下に、このマニュアルの項目を表記されている順に説明します。ほとんどのマニュアルページが下記の項目からなる共通の書式で書かれていますが、必要でない項目については省略されています。たとえば、記述すべきバグがコマンドにない場合などは、「使用上の留意点」という項目はありません。各マニュアルページの詳細は各セクションの intro を、マニュアルページの一般的な情報については man(1) を参照してください。

名前	コマンドや関数の名称と概略が示されています。
形式	<p>コマンドや関数の構文が示されています。標準パスにコマンドやファイルが存在しない場合は、フルパス名が示されます。字体は、コマンド、オプションなどの定数にはボールド体 (bold) を、引数、パラメータ、置換文字などの変数にはイタリック体 (<i>Italic</i>) または <日本語訳> を使用しています。オプションと引数の順番は、アルファベット順です。特別な指定が必要な場合を除いて、1文字の引数、引数のついたオプションの順に書かれています。</p> <p>以下の文字がそれぞれの項目で使われています。</p> <p>[] このかっこに囲まれたオプションや引数は省略できます。このかっこが付いていない場合には、引数を必ず指定する必要があります。</p> <p>... 省略符号。前の引数に変数を付けたり、引数を複数指定したりできることを意味します (例: 'filename.. .')。</p> <p> 区切り文字 (セパレータ)。この文字で分割されている引数のうち1つだけを指定できます。</p> <p>{ } この大かっこに囲まれた複数のオプションや引数は省略できます。かっこ内を1組として扱います。</p>
プロトコル	この項が使われているのは、プロトコルが記述されているファイルを示すサブセクション 3R だけです。パス名は常にボールド体 (bold) で示されています。
機能説明	コマンドの機能とその動作について説明します。実行時の詳細を説明していますが、オプションの説明や使用例はここでは示されていません。対話形式のコマンド、サブコマンド、リクエスト、マクロ、関数などに関しては「使用法」で説明します。
IOCTL	セクション7だけに使用される項です。ioctl(2) システムコールへのパラメータは ioctl と呼ばれ、適切なパラメータを持つデバイスクラスのマニュアルページだけに記載されています。特定のデバイスに関する ioctl は、(そのデバイスのマニュアルページに) アルファベット順に記述されています。デバイスの特定のクラスに関する ioctl は、mtio(7I) のように io で終わる名前が付いているデバイスクラスのマニュアルページに記載されています。
オプション	各オプションがどのように実行されるかを説明しています。「形式」で示されている順に記述されています。オプションの引数はこの項目で説明され、必要な場合はデフォルト値を示します。
オペランド	コマンドのオペランドを一覧表示し、各オペランドがコマンドの動作にどのように影響を及ぼすかを説明しています。
出力	コマンドによって生成される出力 (標準出力、標準エラー、または出力ファイル) を説明しています。

戻り値	値を返す関数の場合、その値を示し、値が返される時の条件を説明しています。関数が 0 や -1 のような一定の値だけを返す場合は、値と説明の形で示され、その他の場合は各関数の戻り値について簡単に説明しています。void として宣言された関数はこの項では扱いません。
エラー	失敗の場合、ほとんどの関数はその理由を示すエラーコードを errno 変数の中に設定します。この項ではエラーコードをアルファベット順に記述し、各エラーの原因となる条件について説明します。同じエラーの原因となる条件が複数ある場合は、エラーコードの下にそれぞれの条件を別々のパラグラフで説明しています。
使用法	この項では、使用する際の手がかりとなる説明が示されています。特定の決まりや機能、詳しい説明の必要なコマンドなどが示されています。組み込み機能については、以下の小項目で説明しています。
	コマンド 修飾子 変数 式 入力文法
使用例	コマンドや関数の使用例または使用方法を説明しています。できるだけ実際に入力するコマンド行とスクリーンに表示される内容を例にしています。例の中には必ず example% のプロンプトが出てきます。スーパーユーザーの場合は example# のプロンプトになります。例では、その説明、変数置換の方法、戻り値が示され、それらのほとんどが「形式」、「機能説明」、「オプション」、「使用法」の項からの実例となっています。
環境	コマンドや関数が影響を与える環境変数を記述し、その影響について簡単に説明しています。
終了ステータス	コマンドが呼び出しプログラムまたはシェルに返す値と、その状態を説明しています。通常、正常終了には 0 が返され、0 以外の値はそれぞれのエラー状態を示します。
ファイル	マニュアルページが参照するファイル、関連ファイル、およびコマンドが作成または必要とするファイルを示し、各ファイルについて簡単に説明しています。
属性	属性タイプとその対応する値を定義することにより、コマンド、ユーティリティ、およびデバイスドライバの特性を一覧しています。詳細は attributes(5) を参照してください。
関連項目	関連するマニュアルページ、当社のマニュアル、および一般の出版物が示されています。

診断	エラーの発生状況と診断メッセージが示されています。メッセージはボールド体 (bold) で、変数はイタリック体 (Italic) または <日本語訳> で示されており、C ロケール時の表示形式です。
警告	作業に支障を与えるような現象について説明しています。診断メッセージではありません。
注意事項	それぞれの項に該当しない追加情報が示されています。マニュアルページの内容とは直接関係のない事柄も参照用に扱っています。ここでは重要な情報については説明していません。
使用上の留意点	すでに発見されているバグについて説明しています。可能な場合は対処法も示しています。

SunOS リファレンスマニュアル (1M) : 保守コマンド

Intro(1M)

名前	Intro, intro – 管理コマンドおよびアプリケーションプログラムの序章
機能説明	<p>本セクションでは、主にシステムの保守や管理に使用するコマンドを、アルファベット順に説明します。</p> <p>コマンドが仮想ファイルシステムのアーキテクチャに合わせて再構成されているため、同じ名前で始まる複数のマニュアルページが存在します。たとえば、mount の名前に関しては、mount(1M)、mount_cachefs(1M)、mount_hfsfs(1M)、mount_nfs(1M)、mount_tmpfs(1M)、mount_ufs(1M) のように 6 つのマニュアルページが存在します。このような場合、最初のマニュアルページにだけ、その総称コマンドの構文およびオプションが説明されています。つまり、これらのオプションは、すべてのファイルシステムのタイプに適用されるということです。以降のマニュアルページには、そのコマンドの機能のうちファイルシステムタイプに特有な部分が説明されています。このようなマニュアルページ名には、下線()とそのコマンドに関係するファイルシステムのタイプ名が伴われています。管理者は、このようなファイルシステムに特有な部分を直接呼び出してはなりません。総称コマンドは、すべてのファイルシステムに共通なインタフェースを提供します。ですから、ファイルシステムタイプに特有なマニュアルページは、個々のコマンドを説明していると考えべきではなく、コマンドのファイルシステムに特有な面を詳細に述べたものだと考えるべきです。</p>
コマンドの構文	<p>特に説明しないかぎり、本セクションで説明するコマンドは、以下の構文に従って、オプションやその他の引数を受け付けます。</p> <pre>name [option(s)] [cmdarg(s)]</pre> <p><i>name</i> 実行可能ファイルの名前です。</p> <p><i>option</i> -noargletter(s) または -argletter<>optarg。</p> <p> <>は、空白 (オプション)。</p> <p><i>noargletter</i> 引数が必要でないオプション 1 文字を表します。</p> <p><i>argletter</i> 引数が必要なオプション 1 文字を表します。</p> <p><i>optarg</i> <i>argletter</i> に必要な引数 (文字列) です。</p> <p><i>cmdarg</i> パス名 (または他のコマンドの引数)。 - だけを指定すると標準入力を表します。</p>
属性	このセクションにリストされた属性については attributes(5) のマニュアルページを参照してください。
関連項目	getopt(1), getopt(3C), attributes(5)
診断	終了時、すべてのコマンドは、正常に終了すると 0 を返します。ゼロでない値を返した場合、間違った引数を指定した、不良で受け入れることができないデータを指定した、その他、現在はうまく処理できないなどの障害を示します。このような値は、「終了コード」、「終了ステータス」、「リターンコード」などさまざまな呼ばれ方をします。そして、特別な使い方がある場合にかぎって説明されます。

注意事項	すべてのコマンドにおいて標準の構文に準拠しているわけではありません。	
コマンド一覧	名前	説明
	accept(1M)	印刷要求の受付または拒否
	acct(1M)	アカウントिंगおよびいろいろなアカウントコマンドの概要
	acctdisk(1m)	acct(1M) を参照
	acctdusg(1m)	acct(1M) を参照
	accton(1m)	acct(1M) を参照
	acctwtmp(1m)	acct(1M) を参照
	afbconfig(1M)	AFB グラフィックスアクセラレータの設定
	catman(1M)	参照マニュアル用のフォーマット整形したファイルの作成
	cfgadm(1M)	構成の管理
	cfgadm_ac(1M)	EXX00 メモリーシステムの管理
	cfgadm_sysctrl(1M)	EXX00 システムボードの管理
	closewtmp(1m)	acct(1M) を参照
	cvcd(1M)	仮想コンソールデーモン
	dd(1M)	ファイルの変換とコピー
	df(1M)	使用可能なディスクブロックおよびファイル数の表示
	du(1M)	ディスク使用状況の要約
	ffbconfig(1M)	FFB グラフィックスアクセラレータの設定
	GFXconfig(1m)	pgxconfig(1M) を参照
	installer(1M)	Solaris Web Start インストールユーティリティ
	lpmove(1M)	印刷要求の移動
	lpsched(1M)	LP 印刷サービスの起動
	lpshut(1M)	LP 印刷サービスの停止
	luxadm(1M)	Sun StorEdge A5000、RSM、SPARCstorage Array (SSA)サブシステム、Sun Fire 880 内部記憶装置サブシステムおよび各 Fiber Channel Arbitrated Loop (FC_AL) デバイスの管理プログラム
	m64config(1M)	M64 グラフィックスアクセラレータの設定
	modinfo(1M)	ロードされているカーネルモジュールについての情報の表示

Intro(1M)

modload(1M)	カーネルモジュールのロード
modunload(1M)	モジュールのアンロード
patchadd(1M)	Solaris 2.x および Solaris 7 システムへのパッチ適用
patchrm(1M)	Solaris 2.x および Solaris 7 システムからのパッチ削除 とパッチ適用前のファイル復元
pgxconfig(1M)	PGX32 (Raptor GFX) グラフィックスアクセラレータの 設定
pmconfig(1M)	電源管理システムの設定
powerd(1M)	電源管理デーモン
prodreg(1M)	Solaris Product Registry の GUI ビューア
prtdiag(1M)	システムの診断情報の出力
raidctl(1M)	RAID ハードウェアユーティリティー
reject(1m)	accept(1M) を参照
restricted_shell(1m)	rsh(1M) を参照
rsh(1M)	制限付きシェルのコマンドインタプリタ
scadm(1M)	システムコントローラ (SC) の管理
su(1M)	スーパーユーザーまたは別のユーザーに変更
SUNWafb_config(1m)	afbconfig(1M) を参照
SUNWffb_config(1m)	ffbconfig(1M) を参照
SUNWm64_config(1m)	m64config(1M) を参照
ttymon(1M)	端末ポートのポートモニター
utmp2wtmp(1m)	acct(1M) を参照
wall(1M)	すべてのユーザーへの通知

名前	accept, reject – 印刷要求の受付または拒否
形式	accept <i>destination</i> ... reject [-r <i>reason</i>] <i>destination</i> ...
機能説明	accept コマンドは、指定した宛先への印刷要求を待ち行列に加えることを許可します。 reject コマンドは、指定した宛先への印刷要求を待ち行列に加えることができないようにします。 宛先が印刷要求を受け付けたか、または拒否したかを確認するには、lpstat -a を実行してください。 accept と request は印刷サーバー上で実行する必要があります。クライアントシステム上では意味をもちません。
オプション	reject コマンドには、次のオプションを指定できます。 -r <i>reason</i> <i>destination</i> への印刷要求を抑止する理由を文字列として記述します。 <i>reason</i> に空白が含まれる場合は引用符で囲ってください。 <i>reason</i> は、プリンタの状況を調べる lpstat -a コマンドの出力中に表示されます。 <i>reason</i> のデフォルト値は、既存の宛先に関しては unknown reason、システムに追加されたばかりでまだ印刷要求を受け付けたことのない宛先に関しては new printer となります。
オペランド	次のオペランドを指定できます。 <i>destination</i> 印刷要求を受け付ける、または拒否する宛先名。宛先にはプリンタ名やプリンタクラスを指定します (lpadmin(1M) 参照)。 <i>destination</i> は名前を使用して指定します。名前の命名規約については printers.conf(4) のマニュアルページを参照してください。
終了ステータス	以下の終了ステータスが返されます。 0 正常終了 0 以外 エラーが発生した
ファイル	/var/spool/lp/* LP 印刷待ち行列
属性	次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWpcu
CSI	対応済み(「注意事項」参照)

accept(1M)

関連項目	enable(1), lp(1), lpstat(1), lpadmin(1M), lpsched(1M), printers.conf(4), attributes(5)
注意事項	<p>accept と reject は印刷サーバーのスプール用システムの待ち行列に対してのみ有効です。したがって、クライアントシステムから出された要求については、印刷サーバーのスプール用システムが取り消したり受け取ったりするまでは、クライアントシステムの印刷待ち行列に入れられたままの状態になります。</p> <p>accept は <i>destinations</i> 名を除いて CSI 対応が可能です。</p>

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre> /usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp </pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

acct(1M)

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、リブートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続いて起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のカラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントリングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントリングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

acctdisk(1M)

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre>/usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp</pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルをセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

acctdisk(1M)

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、レポートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続いて起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のカラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

acctdisk(1M)

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントリングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントリングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre> /usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp </pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

acctdusg(1M)

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、リブートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続いて起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のコラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

acctdusg(1M)

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントリングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントリングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

accton(1M)

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre>/usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp</pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルをセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、レポートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続いて起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のカラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

accton(1M)

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre> /usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp </pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

acctwtmp(1M)

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、リブートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続いて起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のカラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

acctwtmp(1M)

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントリングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントリングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

afbconfig(1M)

名前	afbconfig, SUNWafb_config – AFB グラフィックスアクセラレータの設定
形式	<pre>/usr/sbin/afbconfig [-dev <i>device-filename</i>] [-res <i>video-mode</i> [now try] [noconfirm nocheck]] [-file machine system] [-deflinear true false] [-defoverlay true false] [-overlayorder first last] [-expvis enable disable] [-sov enable disable] [-maxwinds <i>n</i>] [-extovl enable disable] [-g <i>gamma-correction-value</i>] [-gfile <i>gamma-correction-file</i>] [-propt] [-prconf] [-defaults] /usr/sbin/afbconfig [-propt] [-prconf] /usr/sbin/afbconfig [-help] [-res ?]</pre>
機能説明	<p>afbconfig は、AFB グラフィックスアクセラレータおよび AFB 対応の X11 ウィンドウシステムのデフォルトの一部を設定します。</p> <p>afbconfig の次の形式では、指定したオプションを OWconfig ファイルに保存します。</p> <pre>/usr/sbin/afbconfig [-dev<i>device-filename</i>] [-res <i>video-mode</i> [now try] [noconfirm nocheck]] [-file machine system] [-deflinear true false] [-defoverlay true false] [-overlayorderfirst last] [-expvisenable disable] [-sov enable disable] [-maxwinds<i>n</i>] [-extovl enable disable] [-g<i>gamma-correction-value</i>] [-gfile<i>gamma-correction-file</i>] [-propt] [-prconf] [-defaults]</pre> <p>これらのオプションは、次にウィンドウシステムをそのデバイスで実行するときに AFB デバイスを初期化するために使用されます。OWconfig ファイル内のオプションの更新は、異なるウィンドウセッションや再起動したシステムでも有効となります。</p> <p>-prconf、-propt、-help、-res ? オプションだけを起動する次の形式では、OWconfig ファイルは更新されません。</p> <pre>/usr/sbin/afbconfig [-propt] [-prconf] /usr/sbin/afbconfig [-help] [-res ?]</pre> <p>また、次の形式では、その他のオプションはすべて無視されます。</p> <pre>/usr/sbin/afbconfig [-help] [-res ?]</pre> <p>オプションは、一度に 1 つの AFB デバイスに対してのみ指定することができます。複数の AFB デバイスに対してオプションを指定するには、afbconfig コマンドを複数回起動する必要があります。</p> <p>afbconfig で指定できるのは、AFB 固有のオプションだけです。デフォルトの表示色数、デフォルトのビジュアルクラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください。</p>

ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは、/usr/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/etc/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。

これらの標準 OWconfig ファイルのどちらもスーパーユーザーのみが書き込みを行います。したがって、スーパーユーザーが所有する afbconfig プログラムは、setuid による root の権限で実行されます。

オプションのデ
フォルト値

afbconfig コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。ウィンドウシステムを実行する際に、afbconfig による AFB オプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルトは次のとおりです。

```
-dev                /dev/fbs/afb0
-file              machine
-res              none
-deflinear        false
-defoverlay       false
-linearorder      last
-overlayorder     last
-expvis           enabled
-sov              enabled
-maxwids          32
-extovl           enabled
-g                2.22
```

-res オプションのデフォルト値 none とは、ウィンドウシステムが実行された場合に、画面解像度とそのデバイスに現在プログラムされている表示モードになることを意味しています。

これによって、PROM によってデバイスの解像度を指定しているユーザーとの共用性が保てます。(Gx などの)一部のデバイスでは、PROM が表示モードを指定する唯一の手段です。これは、デフォルトの AFB 表示モードは、最終的に PROM によって決まることを意味しています。

オプション

次のオプションがサポートされています。

```
-defaults
  すべてのオプションの値をそれぞれのデフォルト値に戻します。
```

afbconfig(1M)

-deflinear true | false

AFB には、2 種類の画像表示形式があります。リニア画像と非リニア画像です。リニア画像はガンマ補正され、非リニア画像は補正されません。リニア画像版も非リニア画像版も、ともに持つ画像表示形式が 2 つあります。24 ビットトゥルーカラーと 8 ビットスタティックグレーです。

true のときは、デフォルトの画像表示形式として、デフォルトで選択されたオプション (特に、Xsun(1) の defdepth および defclass オプション。詳細は OpenWindows のマニュアルページを参照) を満たすリニア画像がセットされます。

false のとき、または他のデフォルトで選択されたオプションを満たすリニア画像がないときは、これらの他のオプションを満たす、非リニア画像がデフォルトとして選択されます。AFB にはリニアオーバーレイ画像表示形式がないため、-defoverlay オプションが存在するときには、このオプションを使用することはできません。

-defoverlay true | false

AFB が、残りの AFB 画像から切り離されたピクセルを持つ 8 ビット疑似カラー画像を提供します。これを、オーバーレイ画像といいます。この画像表示形式で作成されたウィンドウは、他の画像表示形式で作成されたウィンドウに影響を与えません。逆に、他の画像表示形式で作成されたウィンドウは、オーバーレイウィンドウに影響を与えます。

この画像を使用して作成されたウィンドウで使用することができる色の数は -extov1 オプションの設定に依存します。-extov1 オプションが有効になっている場合は、256 種類の不透明カラーの値による拡張オーバーレイを使用することができます(-extov1 を参照)。-extov1 が無効になっている場合は、拡張オーバーレイを使用することはできず、この画像には、不透明カラーの (256 -maxwids) の値が使用されます (-maxwids を参照)。

-defoverlay の値が true である場合には、オーバーレイ画像がデフォルト画像になります。-defoverlay の値が false の場合には、他のデフォルトで選択された def、depth および defclass オプションを満たすオーバーレイでない画像表示形式が、デフォルトの画像表示形式として選択されます。詳細は、OpenWindows のマニュアルページを参照してください。

-defoverlay true オプションが使用されるときは、常に openwin コマンド行で選択されたデフォルトの深さとクラスは 8 ビット疑似カラーである必要があります。それ以外の場合は、警告メッセージが出力され、-defoverlay オプションは false として扱われます。

-deflinear オプションが存在するときには、AFB にはリニアオーバーレイ画像表示形式がないため、このオプションは使用することができません。

-dev device-filename

AFB 特殊ファイルを指定します。デフォルトは /dev/fbs/afb0 です。

-expvis enable | disable

enable にすると、OpenGL Visual Expansion が起動されます。選択された画像表示形式グループ (8 ビット PseudoColor、24 ビット TrueColor など) が画像表示形式リストに見つかります。

-extovl enable | disable

enable にすると、拡張オーバーレイを使用することができます。このオーバーレイ画像には 256 種類の不透明カラーがあります。SOV 画像には 255 種類の不透明カラーと 1 種類の透明カラーがあります。

また、このオプションは、ハードウェアによる透明カラーを有効にするため、SOV 画像を使用するウィンドウで、より高い性能が得られます。

-file machine | system

更新する OWconfig ファイルを指定します。machine が指定された場合は、/etc/openwin ディレクトリツリーにあるマシン固有の OWconfig ファイルが使用されます。system が指定された場合は、/usr/openwin ディレクトリツリーにある共通の OWconfig ファイルが使用されます。指定されたファイルがない場合は、新たに生成されます。

-g gamma-correction-value

ガンマ補正の値を変更することができます。すべてのリニア画像ではガンマ補正を使用することができます。デフォルトでは、*gamma-correction-value* は 2.22 です。0 より小さい値は無効 (不正) です。ガンマ補正の値はリニア画像に適用され、リニア画像の有効ガンマ値は 1.0 になります。これは、XSolarisGetVisualGamma(3) によって返される値です。この機能については、XSolarisGetVisualGamma(3) を参照してください。

このオプションは、ウィンドウシステムが稼動しているときに使用することができます。ガンマ補正の値を変更すると、リニア画像を使用して表示されているすべてのウィンドウに影響を受けます。

-gfile gamma-correction-file

指定されたファイル (*gamma-correction-file*) からガンマ補正表を読み込みます。*gamma-correction-file* は、各行が R、G、B チャネルの値を持つように書式化されている必要があります。それらの値は、16 進数で指定し、値と値の間は 1 つ以上のスペースで区切ります。また、*gamma-correction-file* は、そのような 3 つの値の組が 256 種類定義されます。

gamma-correction-file の例を以下に示します。

```
0x00 0x00 0x00
0x01 0x01 0x01
0x02 0x02 0x02
...
...
0xff 0xff 0xff
```

このオプションを使用することによって、ウィンドウシステムが稼動しているときにガンマ補正表を読み込むことができます。新しいガンマ補正は、このリニア画像によって表示されているすべてのウィンドウに影響を与えます。ユーザーが指定した表によってガンマ補正を行う際は、ガンマ補正の値は定義されません。デフォルト

afbconfig(1M)

トでは、ウィンドウシステムはガンマ補正值として 2.22 を使用し、このガンマ補正值に対応してウィンドウシステムが作成したガンマ補正表を読み込みます。

-help

afbconfig コマンド行のオプションと機能の概要を一覧で表示します。

-linearorder first | last

first のときには、AFB スクリーン用の X11 スクリーン画像表示形式リスト上で、リニア画像が非リニア画像より前に表示されます。last のときには、非リニア画像は、リニア画像より前に表示されます。

-maxwids *n*

ウィンドウ ID s (WIDs) として使用するために予約される最大数の AFB X チャンネルピクセル値を指定します。オーバーレイカラーマップのピクセル値の残りは、通常の X11 の未使用のカラーピクセルのために使用されます。確保された WIDs は、(XGL などの) 3 次元グラフィックスウィンドウ、MBX ウィンドウと、デフォルト以外の画像表示形式をもつウィンドウにより発生順に割り当てられます。X チャンネルコードの 0 から (255 - *n*) は、未使用のカラーピクセルです。(255 - *n* + 1) から 255 の X チャンネルコードは、WID として使用するために予約されます。適切な値は、1、2、4、8、16、32、64 です。

このオプションは -extov1 が無効になっている場合のみ使用することができます。

-overlayorder first | last

first のときには、AFB スクリーン用の X11 スクリーン画像表示形式リスト上で、8 ビット疑似カラーオーバーレイ画像が、非オーバーレイ画像より前に表示されます。last のときには、非オーバーレイ画像は、オーバーレイ画像より前に表示されます。

-propt

-file オプションで指定された OWconfig ファイルに書かれた AFB オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。afbconfig の呼び出しが終了した後に、OWconfig ファイルに書き込まれるオプションの値を表示します。

次に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/afb0 ---
OWconfig: machine
Video Mode: 1280x1024x76
Default Visual: Non-Linear Normal Visual
Visual Ordering: Linear Visuals are last
                  Overlay Visuals are last
OpenGL Visual Expansion: enabled
Server Overlay Visuals: enabled
Extended Overlay: enabled
Underlay WIDs: 64 (not configurable)
Overlay WIDs: 4 (not configurable)
Gamma Correction Value: 2.220
Gamma Correction Table: Available
```

-prconf

AFB ハードウェア構成を表示します。

次に表示例を示します。

```
--- Hardware Configuration for /dev/fbs/afb0 ---
Type: double-buffered AFB with Z-buffer
Board: rev 0 (Horizontal)
Number of Floats: 6
PROM Information: @(#)afb.fth x.xx xx/xx/xx
AFB ID: 0x101df06d
DAC: Brooktree 9070, version 1 (Pac2)
3DRAM: Mitsubishi 130a, version x
EDID Data: Available - EDID version 1 revision x
Monitor Sense ID: 4 (Sun 37x29cm RGB color monitor)
Monitor possible resolutions: 1024x768x77, 1024x800x84, 1
                             1152x900x76, 1280x1024x67, 1280x1024x76, 960x680xx108s
Current resolution setting: 1280x1024x76
```

-sov enable | disable

enable にすると、ルートウィンドウの `SERVER_OVERLAY_VISUALS` 属性が有効になります。SOV 画像が転送され、それらの透過タイプ、値、階層は、この属性によって参照することができます。**disable** にすると、`SERVER_OVERLAY_VISUALS` 属性は定義されません。SOV 画像は転送されません。

-res video-mode [now | try [noconfirm | nocheck]]

指定した AFB デバイスに接続されているモニターを制御する際に使われる表示モードを指定します。

組み込まれている表示モードの形式は次のとおりです。 `widthxheightxrate width` はピクセル単位のスクリーン幅、`height` はピクセル単位のスクリーンの高さ、`rate` は画面を垂直方向に再描画する周期です。

960x680x112s や 960x680x108s の `s` 接尾辞は、これらが立体表示モードであることを意味します。640x480x60i や 768x575x50i の `i` 接尾辞は、インタレース表示タイミングを有効にします。この接尾辞がない場合は、ノンインタレースタイミングが使用されます。

便宜上、`-res` にリフレッシュレートを指定する際、値の直前に `x` の代わりに `@` を使用できます。たとえば、1280x1024@76 のように指定できます。AFB が対応している一部の表示モードには、モニターが対応していない場合があります。また、AFB がサポートする表示モードにも、モニターがサポートしていないものがあります。AFB デバイスとモニターの両方がサポートしている表示モードのリストは、`-res ?` オプション付きの `afbconfig` (形式の項に記された 3 番目の形式) を実行することによって得ることができます。

AFB がサポートしている表示モードのリストを次に示します。

```
1024x768x60
1024x768x70
1024x768x75
```

afbconfig(1M)

1024x768x77
1024x800x84
1152x900x66
1152x900x76
1280x800x76
1280x1024x60
1280x1024x67
1280x1024x76
960x680x112s (立体表示)
960x680x108s (立体表示)
640x480x60
640x480x60i (インタレース)
768x575x50i (インタレース)

便宜上、AFB がサポートしている表示モードのいくつかには記号名が定義されています。*widthxheightxrate* の形式の代わりに、記号名を *-res* オプションの引数として指定することができます。記号名 *none* は、ウィンドウシステムを実行すると、画面の解像度は現在デバイスにプログラムされている表示モードになることを意味します。

AFB がサポートしている表示モードのリストを次に示します。

記号名 対応する表示モード

svga 1024x768x60
1152 1152x900x76
1280 1280x1024x76
stereo 960x680x112s
ntsc 640x480x60i
pal 768x575x50i
none (上記参照)

-res オプションには、表示モードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定できます。

noconfirm *-res* オプションを指定した際に、システムが使用不可であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために *afbconfig* のデフォルトの動作では、この問題についての警告メッセージと、処理を継続するかどうかを確認するメッセージを表示します。*noconfirm* オプションを指定すると、*afbconfig* コマンドはこの確認をせずに、要求のあった表示モードにプログラムします。このオプションは、*afbconfig* がシェルスクリプトから実行されている場合に便利です。

nocheck このオプションを指定すると、モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定された表示モードは、現在接続されているモニターに適切かど

うかにかかわらず受け付けられます。このオプションは、AFB デバイスに異なるモニターを接続する場合に便利です。このオプションの指定は、noconfirm の指定も兼ねます。

now OWconfig ファイルの表示モードを更新するとともに、AFB デバイスが指定した表示モードにただちにプログラムされます。この機能は、ウィンドウシステムを開始する前に表示モードを変更する際に便利です。

対象となるデバイスが稼働している間 (たとえば、ウィンドウシステムの稼働中) に、この引数をafbconfig に指定することはお勧めしません。予期しない結果になることもあります。now 引数を指定してafbconfig コマンドを実行する場合は、最初にウィンドウシステムを終了してください。now 引数がウィンドウシステムのセッション中に使用された場合、表示モードはただちに変更されますが、画面の幅や高さはそのセッションが終了して次のセッションに入るまで変更されません。さらに、立体表示モードではシステムが変更を認識しないことがあります。したがって、ウィンドウシステムの稼働中には絶対に now オプションを指定しないでください。

try このオプションを指定すると、指定した表示モードが試験的にプログラムされます。ユーザーは、指定した表示モードを使用する場合は、メッセージが表示されてから 10 秒以内に「y」を入力します。表示されたモードを使用しない場合は、10 秒以内に任意の文字を入力します。「y」またはReturn キー以外の文字の入力は、すべて「使用しない」と判断され、以前の表示モードに戻されるため、OWconfig ファイル中の表示モードは書き換えられません (その他の指定されたオプションは有効となります)。Return キーの入力があった場合は、新しい表示モードを保持するかどうかを yes または no で確認するメッセージが表示されます。このオプションの指定は、now 引数の指定も兼ねます (now 引数の注意を参照)。

使用例 例 1 モニターの種類の変更

モニターの種類を、垂直周波数 76 Hz で解像度 1280x1024 に変更する例を以下に示します。

```
example% /usr/sbin/afbconfig -res 1280x1024x76
```

属性 次の属性については、attributes(5) を参照してください。

属性タイプ	属性値
使用条件	SUNWafbcf

afbconfig(1M)

関連項目 | mmap(2), attributes(5)

名前	catman – 参照マニュアル用のフォーマット整形したファイルの作成
形式	<code>/usr/bin/catman [-c] [-n] [-p] [-t] [-w] [-M directory] [-T macro-package] [sections]</code>
機能説明	<p>catman は nroff(1) または sgml(5) の入力ファイルから、オンラインマニュアルページのプレフォーマット・バージョンを生成します。プレフォーマットされたマニュアルページのディレクトリは、自己包括的にかつフォーマットされていないエントリから独立して作成されるので、一群のマシンの間でそれらのマニュアルページを (rdist(1) などによって) 容易に配布することができます。</p> <p>catman は、MANPATH または -M で指定されたディレクトリに windex のデータベースファイルも作成します。windex のデータベースファイルは、キーワード、そのキーワードが指す参照マニュアルページ、参照マニュアルページに書かれたユーティリティまたはインタフェースの目的を説明するテキスト行の 3 つのカラムで構成されているリストです。各キーワードは、「名前」(NAME) の行の ' ' (ダッシュ) の前にあるコンマで区切られた単語のリストから抽出します。キーワードが指す参照マニュアルページは、「名前」の行の最初の単語です。3 つ目のカラムの記述は、「名前」の行の - のあとに続くテキストから抽出します。「名前」の行は、.TH マクロによって作成されるページヘッダーのすぐあとに置く必要があります (必要とされる形式については、「注意事項」を参照)。</p> <p>各マニュアルページを検査し、それに対応するプレフォーマット・バージョンが存在しないものあるいは、現状のマニュアルページよりも古いものについて、プレフォーマット・バージョンを再生成します。変更箇所があれば、catman は windex データベースも再生成します。</p> <p>マニュアルページがシャドウページである場合、つまりその内容が書かれた別のマニュアルページをソースファイルにしている場合、対象となるプレフォーマットされたマニュアルページへのシンボリックリンクが catx または fmtx ディレクトリ内に作成されます。フォーマットされていない nroff のソースファイルにあるシャドウファイルは、最初の行に .so manx/yyy.x という形式の行があることで識別できます。</p> <p>SGML のソースファイルにあるシャドウファイルは、文字列 SHADOW_PAGE があることで識別できます。シャドウファイルで宣言されたファイルの実体は、ソースとなるファイルを示しています。</p>
オプション	<p>以下のオプションを指定できます。</p> <ul style="list-style-type: none"> -c SGML のソースファイルから、フォーマットされていない nroff のソースファイルを適切な man サブディレクトリに作成します。このオプションは SGML ファイルと同じ名前で man ディレクトリにある既存のファイルをすべて上書きします。 -n windex データベースの作成または再生成を行いません。このオプションを指定した場合は windex データベースが作成されないのので、apropos、whatis、man -f、man -k コマンドを実行しても失敗します。 -p 実際の処理は実行せず、どのような処理が行われるかの表示だけを行います。

catman(1M)

	-t	cat サブディレクトリへの nroff を行う代わりに、該当する fmt サブディレクトリ内に troff 処理後のエントリを作成します。
	-w	whatis(1) および man(1) の -f と -k オプションで使用する windex データベースの作成だけを行います。マニュアルページの再フォーマットは実行しません。
	-M <i>directory</i>	引数に指定したディレクトリ (デフォルトは /usr/share/man) 中にあるマニュアルページを更新します。コンマは、マニュアルセクションの番号を示すために使用されているので、-M オプションの引数として指定するディレクトリ名には、'/' (コンマ) が含まれないようにしてください (man(1) 参照)。
	-T <i>macro-package</i>	標準のマニュアルページマクロの代わりに、 <i>macro-package</i> 引数で指定したマクロパッケージを使用します (デフォルトは man(5))。
オペランド		次のオペランドを使用できます。
	<i>sections</i>	先頭文字が '-' でないパラメタは、catman によって処理されるマニュアルセクションとみなします。各セクションは空白で区切ります。このオペランドが指定されると、指定するマニュアルセクションだけが処理されます。次に例を示します。 catman 1 2 3 上記のコマンドは、セクション 1、2、3 のマニュアルページだけを更新します。セクションが指定されない場合、環境変数 MANPATH に指定された man ディレクトリにあるすべてのセクションが処理されます。
環境	TROFF	-t オプションが指定された場合に用いるフォーマットの名前。この環境変数が設定されていない場合は、troff(1) が用いられます。
	MANPATH	catman と man(1) で処理されるディレクトリの、コロンで区切られたリスト。各ディレクトリは、コンマで区切られたセクションのリストの後に続いて指定できます。この環境変数に値を設定すると、デフォルトのディレクトリ検索パスである /usr/share/man と、デフォルトのセクション検索パスである man.cf を無効にします。また、-M フラグはデフォルトのディレクトリ検索パスを、-s フラグはデフォルトのセクション検索パスを、それぞれ無効にします。
ファイル	/usr/share/man	マニュアルページのデフォルトのディレクトリ
	/usr/share/man/man*/*.*	nroff 入力ファイル (未処理)
	/usr/share/man/sman*/*.*	SGML 入力ファイル (未処理)
	/usr/share/man/cat*/*.*	プレフォーマットされた nroff 処理後のマニュアルページ

catman(1M)

/usr/share/man/fmt*/*. * プレフォーマットされた troff 処理後のマニュアルページ

/usr/share/man/windex 目次とキーワードのデータベース

/usr/lib/makewhatis windex データベース生成用のコマンドスクリプト

/usr/share/lib/tmac/an デフォルトのマクロパッケージ

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWdoc
CSI	対応済み

関連項目 apropos(1), man(1), nroff(1), rdist(1), rm(1), troff(1), whatis(1), attributes(5), man(5), sgml(5)

診断 man? メッセージの冒頭に示されたファイルは、カッコ内のファイルが参照しようとしたが、存在しませんでした。

(.so'ed
from
man?
/YYY.?)
: No
such
file or
directory

target catman では、ディレクトリ /usr/man からの相対パスで示したファイル名のみを参照できます。

in man?
/xxx.?
must be
relative
to
/usr/man

opendir: catman が通常検索するディレクトリのうちの 1 つが見つかりません (軽度の警告メッセージ)。
: No
such
file or
directory

catman(1M)

警告	<p>*.*: No catmanによって、中身が空のディレクトリが見つかりました (軽度の警告メッセージ)。</p> <p>file or directory 以前に catman を実行して cat* ディレクトリがすでにインストールされている場合に、オペレーティングシステムをアップグレードした場合は、catman を実行する前に cat* ディレクトリ構造全体を削除してください (rm(1) 参照)。</p>
注意事項	<p>すべての man* ディレクトリがそろっていない場合には、whatis データベースを再構築するために catman を再度実行しないでください。catman は、man* ディレクトリにもとづいてこの windex ファイルを構築します。</p> <p>catman には、正しい windex のインデックスファイルを生成するための要件があります。catman は、個々のマニュアルページのファイル中に、特定の形式を持つ 2 つのマクロ行 (ページの先頭の .TH 行と .SH NAME の行) を必要とします。</p>
	<p>.TH マクロには、最低 3 つの引数 (ファイル名、セクション番号、日付) が必要です。 .TH 行は、.TH マクロで始まり、そのあとに、1 つの空白、マニュアルページのファイル名、1 つの空白、セクション番号、1 つの空白、日付が続きます。日付は、“day month year” (日本語の場合は “年月日”) として二重引用符で囲みます。このうちの month には 3 文字の省略形 (Jan, Feb, Mar, など) を指定します。</p>
	<p>「名前」(NAME) の行としても知られる .SH NAME マクロは、.TH 行のすぐあとに置く必要があります (これら 2 つの行の間には何も置かない)。「名前」の行では、フォントの変更はできません。.SH NAME のすぐ次の行には、マニュアルページのファイル名、コンマで区切られたシャドウファイル名 (もしあれば)、ダッシュ、簡単な概要を含む行が続きます。これらの要素は、すべて 1 つの行に収める必要があります (改行を入れないでください)。</p>
	<p>.TH 行と .SH NAME の行の正しいコーディング例を次に示します。</p>
	<pre>.TH nismatch 1M "10 Apr 1998" .SH NAME nismatch, nisgrep \- utilities for searching NIS+ tables</pre>

名前 | cfgadm – 構成の管理

形式 | `/usr/sbin/cfgadm [-f] [-y | -n] [-v] [-o hardware_options] -c function ap_id...``/usr/sbin/cfgadm [-f] [-y | -n] [-v] [-o hardware_options] -x hardware_function ap_id...``/usr/sbin/cfgadm [-v] [-a] [-s listing_options] [-o hardware_options] [-l [ap_id | ap_type]]``/usr/sbin/cfgadm [-v] [-o hardware_options] -t ap_id...``/usr/sbin/cfgadm [-v] [-o hardware_options] -h [ap_id | ap_type]`

機能説明

cfgadm コマンドを使用して、動的な再構成が可能なハードウェア資源に対して構成の管理を行うことができます。これらの操作には、状態 (state) の表示 (-l)、検査の開始 (-t)、構成状態の変更の開始 (-c)、ハードウェア固有の機能の実行 (-x)、および構成管理のヘルプ情報の表示 (-h)、が含まれます。構成管理は、接続点 (attachment point) で実行されます。接続点は、Solaris の動作中にハードウェア資源の動的再構成を行うことにシステムソフトウェアが対応している場所です。

構成の管理では、マシン上に実際にあるハードウェア資源と、構成済みで Solaris が認識できるハードウェア資源が区別されます。構成管理機能の特性はハードウェアに依存し、ハードウェア固有のライブラリを呼び出すことで実行されます。

構成管理は、接続点で実行されます。接続点に設置されているハードウェア資源には、システムの稼働中に物理的な交換ができるものとできないものがありますが、構成管理インタフェースによって、動的に再構成することはできません。

接続点は、接続点の向こう側に位置するハードウェア資源とは別の2つの固有の要素を定義します。接続点の2つの要素は、受容体 (receptacle) と占有装置 (occupant) です。ハードウェア資源の物理的な取り付け、取り外しは接続点で行われ、その結果、受容体に占有装置が追加されたり削除されたりします。構成管理は、接続点での構成管理機能だけでなく、この物理的な着脱操作にも対応しています。

接続点には、状態 (state) と条件 (condition) の情報が関連付けられています。構成管理インタフェースを使用して、接続点の状態の変化を制御することができます。受容体は、empty、disconnected、connected の3つの状態のいずれかになります。また、占有装置は、configured と unconfigured のいずれかの状態になります。

受容体は、接続点に占有装置がない場合に、必ず受容体の通常の状態である empty になります。この状態には、稼働中のシステムの一部を一時的に停止することができます。ハードウェア固有の機能が関係します。受容体が占有装置をシステムの通常の使用から切り離すことができる場合に、その受容体は disconnected 状態になることもできます。この状態は、占有装置のハードウェア資源をシステムが完全に利用できるようにする前にそのハードウェアに対する検査を実行する場合や、占有装置の物理的な取り外しや再構成のための準備の1つの段階として、主に使用されます。disconnected 状態の受容体は、ハードウェアの許容範囲内で占有装置をシステムから分離しますが、検査や設定が必要な場合は使用を許可する場合があります。受容体

は、占有装置に含まれるハードウェア資源の通常の使用が許可されている場合に、必ず `connected` 状態になります。`connected` 状態は、占有装置を含み、かつ構成管理操作が実行されていない受容体の通常の状態です。

`unconfigured` 状態の占有装置に含まれるハードウェア資源は、Solaris の通常のデータ構造では表現されないため、Solaris はそのハードウェアを使用できません。未構成の占有装置に対して実行できる操作は、構成管理操作に限られています。`configured` 状態の占有装置に含まれるハードウェア資源は、Solaris の通常のデータ構造で表現されるため、Solaris は、一部またはすべてのハードウェア資源を使用することができます。占有装置は、必ず `configured` 状態か `unconfigured` 状態になります。

接続点は、`unknown`、`ok`、`failing`、`failed`、`unusable` の 5 つの条件のいずれかになります。接続点は、電源投入検査と不揮発性記録保存の結果によって、システムをどの条件にも置く可能性があります。

`configured` 状態の占有装置を持つ接続点は、`unknown`、`ok`、`failing`、`failed` の条件のいずれかになります。`failing` または `failed` 条件にない接続点は、ハードウェア固有の回復可能なエラーがしきい値を超えると、操作中に `failing` 状態になる場合があります。また、`failed` 条件にない接続点は、回復不可能なエラーによって、操作中に `failed` 条件に変わる場合があります。

`unconfigured` 状態にある占有装置を持つ接続点は今までに挙げた条件のどれになる可能性もあります。`unconfigured` 状態にある占有装置を持つ接続点の条件は、マシン固有の時間しきい値が経過した後、`ok` から `unknown` になる場合があります。検査機能を開始した場合は、検査の結果によって接続点の条件が `ok`、`failing`、`failed` のいずれかに変わります。検査機能を持たない接続点は、接続点を `unknown` 条件のままにする場合があります。検査が中断された場合は、接続点の条件は、以前の条件、`unknown`、`failed` に設定することができます。`unknown`、`ok`、`failing`、`failed` のいずれかの条件にある接続点には、再検査を行うことができます。

接続点は、さまざまな理由によって `unusable` 条件になります。理由としては、受容体に対する不適切な電力投入や冷却、占有装置が認識できない、対応していない、不適切に構成されている、などが挙げられます。`unusable` 条件にある接続点は、システムで使用することができません。通常、この条件は、接続点に対して物理的な対処がなされない限り変わりません。

また、接続点は、状態の変更処理が進行中である場合や、条件が再評価されている場合に、それを示す使用状態情報を保持します。

接続点は、システムデバイス階層構造の中での接続点のタイプと位置に関連するハードウェア固有の識別子 (`ap_ids`) に対応しています。`ap_id` は単一の接続点を特定するために、一意になっている必要があります。`ap_id` の仕様には、物理タイプと論理タイプの 2 種類が用意されています。物理 `ap_id` には、完全なパス名を指定します。論理 `ap_id` には、簡略表記法を使用し、ユーザーにとってより簡単な方法で接続点を指定します。

たとえば、システムのバックプレーンスロット番号 7 にある接続点の物理 *ap_id* は `/devices/central/fhc/sysctrl:slot7` となり、論理 *ap_id* は `system:slot7` になります。また、システムの第 2 PCI 入出力バス上にある 3 番目の受容体の論理 *ap_id* は `pci2:plug3` になります。

接続点も動的に作成されます。動的接続点には、そのシステムに設定されている基本接続点を基にして名前が付けられます。動的接続点の *ap_ids* は、2 つのコロン、基本構成要素、および動的構成要素で構成されます。基本構成要素は、基本接続点 *ap_id* です。動的構成要素は、ハードウェア固有で、対応するハードウェア固有のライブラリによって生成されます。

たとえば、SCSI HBA を表現し、物理 *ap_id* が `/devices/sbus@1f,0/SUNW,fas@e,8800000:scsi` で、論理 *ap_id* が `c0` である基本接続点を想定します。この SCSI HBA に接続されているディスクは、論理 *ap_id* が `c0::dsk/c0t0d0` である動的接続点によって表現されます。ここで、`c0` は基本構成要素で、`dsk/c0t0d0` はハードウェア固有の動的構成要素です。同様に、この動的接続点の物理 *ap_id* は `/devices/sbus@1f,0/SUNW,fas@e,8800000:scsi::dsk/c0t0d0` になります。

ap_type は *ap_id* の一部で、それ自身だけでは一意にならず、単一の接続点を特定することができません。*ap_type* は、論理 *ap_id* の一部を含み、コロン (:) 区切り記号を含まない部分文字列です。たとえば、`pci` の *ap_type* は、論理 *ap_id* が `pci` で始まる接続点をすべて出力します。

ap_types は、できるだけ使用しないでください。-s オプションの新しい選択サブオプションを使用すれば、より汎用的な方法で柔軟に接続点を選択することができます。【オプション】を参照してください。

cfgadm コマンドは、ハードウェア固有ライブラリに含まれるハードウェア固有の機能と主に対話するため、cfgadm コマンドの動作はハードウェアに依存します。

それぞれの構成管理操作では、サービスの中断が必要になる場合があります。要求された機能を完了するために、対話式で操作中のユーザーにとって目に見えるサービスの中断が必要になる場合は、機能の開始前に標準エラー出力に確認メッセージが表示され、標準入力による確認を促します。すべての質問に対する `yes` を意味する -y オプション、または `no` を意味する -n オプションを指定することによって、確認を省略することもできます。検査レベルなどのハードウェア固有のオプションは -o オプションを使用して、サブオプションとして指定することができます。

システム構成の状態を変更する操作は、システムログデーモンである `syslogd(1M)` によって監視されます。

このコマンドの引数は、`getopt(3C)` および `getsubopt(3C)` の構文規約に従います。

オプション 以下のオプションを使用することができます。

cfgadm(1M)

- a
-l オプションによって、動的接続点のリストも出力されるように指定します。
- c *function*
ap_id で指定された接続点の状態を *function* で指定された状態に変更します。
- function* には、insert、remove、disconnect、connect、configure、unconfigure のいずれかを指定することができます。これらの関数は、ハードウェア固有のライブラリルーチン呼び出しで接続点の状態を変更します。これらの関数の定義を以下に示します。
- | | |
|-------------|--|
| insert | 占有装置を手動で追加する操作を実行したり、物理的な追加を実行するハードウェア機能を起動します。insert には、システムの一部を一時的に停止するハードウェア固有の副作用が伴う場合があります。このような場合、ハードウェア固有のライブラリは、対応する警告メッセージを生成し、ユーザーに対して、そのハードウェア固有の問題と手順を提供します。ハードウェアに起因するさまざまなエラーによってこの関数が失敗し、受容体の条件が unusable になる場合があります。 |
| remove | 占有装置を手動で削除する操作を実行したり、物理的な削除を実行するハードウェア機能を起動します。remove には、システムの一部を一時的に停止するハードウェア固有の副作用が伴う場合があります。このような場合、ハードウェア固有のライブラリは、対応する警告メッセージを生成し、ユーザーに対して、そのハードウェア固有の注意事項と手順を提供します。ハードウェアに起因するさまざまなエラーによってこの関数が失敗し、受容体の条件が unusable になる場合があります。 |
| disconnect | ハードウェア固有の操作を実行して、受容体を disconnected 状態にします。disconnected 状態にすることによって、占有装置に対して、受容体を介した通常の方法による操作が行えなくなります。 |
| connect | ハードウェア固有の操作を実行して、受容体を connected 状態にします。受容体が connect 状態にすることによって、占有装置に対して、受容体を介した通常の方法による操作が行えるようになります。 |
| configure | ハードウェア固有の操作を実行して、占有装置のハードウェア資源を Solaris が使用できるようにします。構成された占有装置はシステム構成の一部になり、psradm(1M)、mount(1M)、ifconfig(1M) などの Solaris デバイス操作メンテナンスコマンドによる操作の対象となります。 |
| unconfigure | ハードウェア固有の操作を実行して、占有装置のハードウェア資源をシステムから論理的に削除します。この関数を使用するには、占有装置が現在構成されていて、占有装置のハードウェアが Solaris によって使用されていない必要があります。 |

状態変更関数は、接続点の条件や、その他のハードウェア固有の問題によって失敗する場合があります。資源を追加するための状態変更関数 (`insert`、`connect`、`configure`) は、接続点が `ok` または `unknown` 条件にある場合に、ハードウェア固有のライブラリに渡されます。接続点がそれ以外の条件にある場合は、強制オプション (`-f`) を使用した場合に限って、資源を追加するための関数がハードウェア固有のライブラリに渡されます。システムからハードウェア資源を削除するための関数 (`remove`、`disconnect`、`unconfigure`) によるハードウェア固有のライブラリの呼び出しは、接続点の条件によって妨げられることはありません。接続点が `unknown` 条件にある場合に、関数は、ハードウェア固有のライブラリによって拒否される場合があります。

接続点の条件は、状態変更関数によって変更されないこともありますが、状態変更操作中のエラーによって接続点の条件が変わる場合があります。条件の書き換えと状態の強制的な変更は、強制オプション (`-f`) を指定した場合だけ実行することができます。これらの処理は、強制オプションを指定しないと失敗します。強制オプションは、ハードウェア固有の安全性検査および完全性検査によって無効になる場合があります。

`-f`

指定された処理を強制的に実行します。これは主に、ハードウェア固有の安全機能を無効にするために使用します。状態の変更操作を強制することによって、ハードウェア固有の安全検査によって `ok` や `unknown` 状態にない占有装置のハードウェア資源を使用することができる場合があります。

`-h [ap_id | ap_type ...]`

ヘルプメッセージテキストを出力します。`ap_id` または `ap_type` を指定すると、この引数によって指定された接続点に関するハードウェア固有のライブラリのヘルプルーチンが呼び出されます。

`-l [ap_id | ap_type ...]`

指定された接続点の状態や条件を一覧表示します。接続点を抽出するには、`-s` オプションと `select` サブオプションを使用します。いずれかの処理オプションを使用せずに `cfgadm` コマンドを起動するのは、引数を使用せずに `-l` を使用することと同じです。表示画面の書式は `-v` および `-s` オプションによって制御されます。`-a` オプションが指定されているときは、接続点が動的に展開されます。

`-n`

対話型の確認を行わず、応答が `no` であるとみなします。`-n` と `-y` のいずれも指定しないと、標準エラー出力と標準入力によって対話型の確認が行われます。これらの標準的なチャンネルのいずれも端末 (`isatty(3C)` によって判定されている) に対応していない場合は、`-n` オプションが想定されます。

`-o hardware_options`

コマンドの主オプションに対してハードウェア固有のオプションを指定します。`hardware_options` の文字列の書式と内容は完全にハードウェア固有のものです。オプション文字列の `hardware_options` は `getsubopt(3C)` の構文規約に従います。

`-s listing_options`

一覧表示 (`-l`) コマンドに対して一覧表示オプションを指定します。`listing_options` は `getsubopt(3C)` の構文規約に従います。サブオプションを使用して、接続点の選択条件 (`select=select_string`)、適切な照合タイプ (`match=match_type`)、一覧表

cfgadm(1M)

示する順序 (*sort=field_spec*)、表示するデータ (*cols=field_spec* と *cols2=field_spec*)、列の区切り記号 (*delim=string*)、列の見出し行の抑制 (*noheadings*) を指定します。

select サブオプションを指定すると、指定された条件と一致する接続点だけが一覧表示されます。*select* サブオプションの構文は次のとおりです。

```
cfgadm -s select=attr1(value1):attr2(value2)...
```

attr は、*ap_id*、*class*、*type* のいずれかです。*ap_id* は論理 *ap_id* フィールド、*class* は接続点のクラス、*type* はタイプフィールドです。*value1*、*value2* などは、照合する値です。照合タイプを指定するには、次のように *match* サブオプションを使用します。

```
cfgadm -s match=match_type,select=attr1(value1)...
```

match_type は、*exact* または *partial* から選択します。デフォルト値は *exact* です。

select サブオプションの引数は、シェルから保護するために引用符で囲みます。

field_spec には、1つの *data-field* または *data-field:data-field:data-field* のようにコロンの(:)で区切った複数の *data-field* を指定します。*data-field* は、*ap_id*、*physid*、*r_state*、*o_state*、*condition*、*type*、*busy*、*status_time*、*status_time_p*、*class*、*info* のいずれかです。*ap_id* フィールドの出力は接続点の論理名で、*physid* フィールドは物理名です。*r_state* フィールドは、*empty*、*disconnected*、*connected* のいずれかになります。*o_state* フィールドは、*configured* と *unconfigured* のいずれかになります。*busy* フィールドは、接続点が使用中の場合に *y* になり、使用中でない場合に *n* となります。*type* フィールドと *info* フィールドはハードウェア固有のフィールドです。*status_time_p* フィールドは、構文解析が可能な *status_time* フィールドです。接続点にクラスが関連付けられている場合は、*class* フィールドにクラス名が表示されます。接続点にクラスが関連付けられていない場合は、*class* フィールドに *none* が表示されます。

field_spec 内のフィールドの順序は重要です。*sort* サブオプションでは、最初に与えられたフィールドが主ソートキーになります。*cols* および *cols2* サブオプションでは、指定した順序でフィールドが出力されます。*cols* 内のフィールドは常に出力されます。*cols2* 内のフィールドは、指定された場合に限り2行目に出力されます。*data-field* に対するソートの順序は、*sort* サブオプションに対する *field_spec* 内の *data-field* 名の前にマイナス(-)を付けることによって逆になります。*sort* のデフォルトの値は *ap_id* です。*cols* および *cols2* のデフォルトの値は *-v* オプションが指定されているかどうかによって異なります。*-v* が指定されていない場合は、*cols* は *ap_id:r_state:o_state:condition* になり、*cols2* は設定されません。*-v* が指定されている場合は、*cols* は *ap_id:r_state:o_state:condition:info* になり、*cols2* は *status_time:type:activity:physid* になります。*delim* のデフォルトの値は、単一の空白文字です。*delim* の値には、任意の長さの文字列を指定することもできます。区切り記号にはコンマ(,)を含めることはできません(*getsubopt(3C)* を参照)。これらの一覧表示オプションは、構文解析が可能な出力を生成するために使用することができます。「注意事項」を参照してください。

-t

1 つまたは複数の接続点の検査を実行します。この検査機能は、接続点の条件を再評価するために使用します。*hardware_options* の中で検査レベル指示子を指定しないと、重度の障害を検出する最も早い検査方法が使用されます。

より包括的な検査は個々のハードウェアに依存するため、*hardware_options* を使用して選択します。

検査の結果は、指定された占有装置の条件を >ok (障害が発見されなかった場合)、*failing* (回復可能な障害が発見された場合)、*failed* (回復不可能な障害が発見された場合) のいずれかに更新するために使用されます。

検査が中断された場合は、接続点の条件は、以前の値に戻るか、ok (障害が発見されなかった場合)、*failing* (回復可能な障害が発見された場合)、*failed* (回復不可能な障害が発見された場合) のいずれかに設定されます。接続点は、エラーがなく、検査が正常に完了した場合のみ ok に設定されます。

-v

詳細表示モードで実行します。-c、-t、-x オプションを指定した場合に、各試行操作の結果を表示するメッセージを出力します。-h オプションを指定した場合は、詳細なヘルプ情報が表示されます。-l オプションを指定した場合は、各接続点に関するすべての情報を出力します。

-x*hardware_function*

ハードウェア固有の機能を実行します。受容体や占有装置の状態は、専用ハードウェア固有の機能を使用して変更します。接続点の状態は、ハードウェア固有の機能の動作中に検出されたエラーの結果として変化する場合があります。

hardware_function 文字列の書式と内容は完全にハードウェア固有のものです。オプション文字列の*hardware_function* は `getsubopt(3C)` の構文規約に従います。

-y

対話型の確認を行わず、応答が *yes* であるとみなします。

使用方法

このコマンドを使用するために必要な特権は、ハードウェアに依存します。一般的に、デフォルトのシステム設定では、一覧表示オプション以外のすべての機能はスーパーユーザーのみが使用することができます。

使用例

例 1 デバイスツリー内の接続点の一覧表示

以下の例は、動的接続点以外のすべての接続点を一覧表示します。

```
example# cfgadm
```

Ap_Id	Type	Receptacle	Occupant	Cond
system:slot0	cpu/mem	connected	configured	ok
system:slot1	sbus-upa	connected	configured	ok
system:slot2	cpu/mem	connected	configured	ok
system:slot3	unknown	connected	unconfigured	unknown
system:slot4	dual-sbus	connected	configured	failing
system:slot5	cpu/mem	connected	configured	ok
system:slot6	unknown	disconnected	unconfigured	unusable
system:slot7	unknown	empty	unconfigured	ok
c0	scsi-bus	connected	configured	unknown

例 1 デバイスツリー内の接続点の一覧表示 (続き)

```
c1          scsi-bus      connected    configured    unknown
```

例 2 構成することができるすべてのハードウェアの情報の一覧表示

以下の例は、現在構成することができるすべてのハードウェアの情報を一覧表示します。動的接続点で表現されるハードウェアも表示されます。

```
example# cfgadm -al
```

Ap_Id	Type	Receptacle	Occupant	Cond
system:slot0	cpu/mem	connected	configured	ok
system:slot1	sbus-upa	connected	configured	ok
system:slot2	cpu/mem	connected	configured	ok
system:slot3	unknown	connected	unconfigured	unknown
system:slot4	dual-sbus	connected	configured	failing
system:slot5	cpu/mem	connected	configured	ok
system:slot6	unknown	disconnected	unconfigured	unusable
system:slot7	unknown	empty	unconfigured	ok
c0	scsi-bus	connected	configured	unknown
c0::dsk/c0t14d0	disk	connected	configured	unknown
c0::dsk/c0t11d0	disk	connected	configured	unknown
c0::dsk/c0t8d0	disk	connected	configured	unknown
c0::rmt/0	tape	connected	configured	unknown
c1	scsi-bus	connected	configured	unknown

例 3 接続点の属性に基づいた選択的一覧表示

以下の例は、接続点のうち、scsi で始まるクラス、c で始まる ap_id、および scsi で始まる type フィールドを持つものをすべて一覧表示します。-s オプションの引数は、シェルから保護するために引用符で囲みます。

```
example# cfgadm -s "match=partial,select=class(scsi):ap_id(c):type(scsi)"
```

Ap_Id	Type	Receptacle	Occupant	Cond
c0	scsi-bus	connected	configured	unknown
c1	scsi-bus	connected	configured	unknown

例 4 現在構成することができるハードウェアの情報の詳細表示

以下の例は、ap-type system の現在構成することができるハードウェアの情報を詳細表示モードで示します。

```
example# cfgadm -v -l system
```

Ap_Id	Type	Receptacle	Occupant	Condition	Information
When	Type	Busy	Phys_Id		
system:slot1			connected	configured	ok
Apr 4 23:50	sbus-upa	n	/devices/central/fhc/sysctrl:slot1		
system:slot3			connected	configured	ok non-detachable
Apr 17 11:20	cpu/mem	n	/devices/central/fhc/sysctrl:slot3		
system:slot5			connected	configured	ok
Apr 4 23:50	cpu/mem	n	/devices/central/fhc/sysctrl:slot5		
system:slot7			connected	configured	ok

例 4 現在構成することができるハードウェアの情報の詳細表示 (続き)

```
Apr  4 23:50 dual-sbus n          /devices/central/fhc/sysctrl:slot7
```

例 5 ハードウェア固有の拡張検査

以下の例は、ハードウェア固有の拡張検査を使用して 2 つの占有装置を検査します。

```
example# cfgadm -v -o extended -t system:slot3 system:slot5
Testing attachment point system:slot3 ... ok
Testing attachment point system:slot5 ... ok
```

例 6 強制オプション

以下の例は、強制オプションを使用して、failing 状態の占有装置をシステムに構成します。

```
example# cfgadm -f -c configure system:slot3
```

例 7 システムから占有装置を構成解除する

以下の例は、システムから占有装置を構成解除します。

```
example# cfgadm -c unconfigure system:slot4
```

例 8 接続点の占有装置を構成する

以下の例は、占有装置を構成します。

```
example# cfgadm -c configure c0::dsk/c0t0d0
```

環境 cfgadm の実行に影響を及ぼす環境変数の LC_TIME、LC_MESSAGES、NLSPATH、TZ については、environ(5) を参照してください。

LC_MESSAGES cfgadm が見出し行とエラーメッセージを表示する方法を特定しません。出力されるデータは、この環境変数の影響を受けません。

LC_TIME cfgadm が状態の変更時間 (status_time) を表示する方法 (人による判読が可能な形式) を特定します。

TZ 状態の変更時間を変換する際に使用される時間帯を特定します。これは人による判読が可能な (status_time) と構文解析が可能な (status_time_p) 形式の両方に当てはまります。

終了ステータス 以下の終了値が返されます。

- 0 正常に完了しました。
- 1 エラーが発生しました。
- 2 指定された宛先が構成管理に対応していません。
- 3 使用方法上のエラー。

属性 以下の属性については、attributes(5) を参照してください。

cfgadm(1M)

属性タイプ	属性値
使用条件	SUNWcsu

関連項目 `cfgadm_pci(1M)`, `cfgadm_sbd(1M)`, `cfgadm_scsi(1M)`, `ifconfig(1M)`, `mount(1M)`, `psradm(1M)`, `prtdiag(1M)`, `syslogd(1M)`, `config_admin(3CFGADM)`, `getopt(3C)`, `getsubopt(3C)`, `isatty(3C)`, `attributes(5)`, `environ(5)`

診断 診断メッセージは標準エラー出力に表示されます。オプションや使用方法のエラー以外に、次の診断メッセージがこのユーティリティによって表示されます。

```
cfgadm: Configuration administration not supported on ap_id
cfgadm: No library found for ap_id
cfgadm: ap_id is ambiguous
cfgadm: operation: Insufficient privileges
cfgadm: Attachment point is busy, try again
cfgadm: No attachment points with specified attributes found
cfgadm: System is busy, try again
cfgadm: operation: Operation requires a service interruption
cfgadm: operation: Data error: error_text
cfgadm: operation: Hardware specific failure: error_text
```

エラーメッセージの詳細については `config_admin(3CFGADM)` を参照してください。

注意事項 ハードウェア資源は、ハードウェア固有の方法で未構成プールに入ります。これは、システムの初期設定や構成解除操作の結果としてなど、さまざまな状況で発生します。unconfigured 状態にある占有装置は、システムによる特定の介入が発生するまで、システムが使用することはできません。このような干渉は、オペレーターが起動したコマンドや、自動構成機構によって発生します。

`cfgadm` コマンドの一覧表示オプションは、シェルスクリプトの中などで、別のコマンドに対する構文解析可能な入力として使用することができます。構文解析可能な出力の場合は、必要なフィールドを選択するときに、`-s` オプションを使用する必要があります。`-s` オプションは、列の見出しを抑制するときにも使用できます。`ap_id`, `physid`, `r_state`, `o_state`, `condition`, `busy`, `status_time_p`, `class`, `type` フィールドは、常に構文解析が可能な出力を生成します。

以下はシェルスクリプトの一部で、タイプが CPU の正常な未構成の占有装置で最初のものを検出します。

```
found=
cfgadm -l -s "noheadings,cols=ap_id:r_state:condition:type" | \
while read ap_id r_state cond type
do
```

```
if [ "$r_state" = unconfigured -a "$cond" = ok -a "$type" = CPU ]
then
    if [ -z "$found" ]
    then
        found=$ap_id
    fi
fi
done
if [ -n "$found" ]
then
    echo "Found CPU $found"
fi
```

構文解析が可能な時間フィールド (`status_time_p`) の形式は、`YYYYMMDDhhmmss` で、文字列の比較を行うのに都合がよい書式で、年、月、日、時間、分、秒を表示します。

システム構成管理で使用できるものの詳細については、ハードウェア固有のマニュアルを参照してください。

cfgadm_ac(1M)

名前	cfgadm_ac - EXX00 メモリーシステムの管理
形式	<pre>/usr/sbin/cfgadm [-c configure] [-f] [-o disable-at-boot enable-at-boot] ac#:bank# ... /usr/sbin/cfgadm [-c unconfigure] [-o disable-at-bootp enable-at-boot] ac#:bank# ... /usr/sbin/cfgadm [-v] [-o quick normal extended, [max_errors=#]] -t ac#:bank#... /usr/sbin/cfgadm -x relocate-test ac#:bank# ... /usr/sbin/cfgadm [-l] -o disable-at-boot enable-at-boot ac#:bank# ...</pre>
機能説明	<p>ac ハードウェア固有ライブラリ</p> <p><code>/usr/platform/sun4u/lib/cfgadm/cfgadm_ac.so.1</code> は、<code>cfgadm_sysctrl(1M)</code> を使用する CPUU/メモリーボードの Dynamic Reconfiguration (DR: 動的再構成) の一部として、E6X00、E5X00、E4X00、E3X00 システムのメモリーバンクの構成と構成解除に関連する機能を提供します。</p> <p>メモリーバンクは、デバイスツリー上の接続点として表されます。CPU/メモリーボードそれぞれに2つの接続点が用意されます。1つがボード上の各バンク (バンク 0 とバンク 1) 用になります。バンクが空いている場合は、受容体の状態は <code>empty</code> (空き) になります。バンクが使用されている場合は、受容体の状態は <code>connected</code>(接続) になります。受容体の状態が <code>disconnected</code> (接続解除) になることはありません。接続されているメモリーバンクの占有状態は、構成することも構成解除することもできません。占有状態が構成されている場合は、メモリーは Solaris によって使用されています。構成解除されている場合は、使用されていません。</p>
オプション	<p>コマンドオプションの詳細については、<code>cfgadm(1M)</code> を参照してください。</p> <p>以下のオプションがサポートされています。</p> <p><code>-c configure unconfigure</code> 占有状態を変更します。引数が <code>configure</code> の場合は、メモリーが初期化され、Solaris のメモリープールに追加します。引数が <code>unconfigure</code> の場合は、Solaris によって使用されているメモリーを切り離します。CPU/メモリーボードをシステムから取り外すには、2つのメモリーバンクを構成解除する必要があります。</p> <p>ボード上のメモリーが <code>disabled-at-boot</code> (<code>info</code> フィールド参照) に指定されていて、<code>-f</code> (強制) オプションやブート許可フラグ (<code>-o enable-at-boot</code>) がいずれも指定されていない場合は、<code>cfgadm</code> コマンドは、構成操作を許可しません。構成操作には、初期化する必要のあるメモリーのサイズに応じて、少し時間がかかります。</p> <p>システムに十分な使用可能メモリーがない (VM viability エラー)、または構成解除されるべきバンクに切り離せないメモリーがある (<code>non-relocatable pages</code> エラー) 場合、<code>cfgadm</code> コマンドは、構成解除操作を許可しません。再配置不可のページの現状は、<code>info</code> フィールドの一覧にある <code>permanent</code> によって表されます。Solaris が使用しているメモリの取り外しを行うと、システム負荷や補助記憶</p>

装置のページングの大きさによって、かなり時間がかかります。構成解除操作は、いつでも中止することができ、シグナルによるコマンドの中断によって、メモリーは完全に構成されている状態に戻ります。もし、タイムアウトまでの期間内に削除できるメモリーがない場合は、構成解除操作は、自動的に取り消されます。デフォルトのタイムアウト期間は 60 秒ですが、`-o timeout=#` オプションで変更することもできます。この数値を 0 にすると、タイムアウトは不許可になります。

`-f`

強制オプション。このオプションは、非揮発性の変数 `disabled-memory-list` で、`disabled at boot` (ブート不可) に指定された、構成されているメモリーバンクのブロックを無効にしたい場合に使用してください。詳細は、『特記事項: Sun Enterprise 6x00/5x00/4x00/3x00 システム』を参照してください。

`-l`

リストオプション。このオプションについての説明は、`cfgadm(1M)` のマニュアルページを参照してください。

`type` フィールドは、常に `memory` です。

`info` フィールドには、以下に示すような空のメモリーバンクについての情報が保持されます。

`slot# empty`

`slot#` は、CPU/メモリーボードが挿入されているシステムスロットを示しています。たとえば、これがスロット 11 である場合、`cfgadm` が関連付けられたボードを操作するために使用する接続点は、`sysctrl0:slot11` になります。`info` フィールドには、以下に示すような接続されているバンクについての情報が保持されます。

`slot# sizeMB|sizeGB [(sizeMB|sizeGB を使用)] base 0x###`
`[interleaved #-way] [disabled at boot] [permanent]`

バンクのサイズは、MB か GB か適切な単位で与えられます。メモリーがまだ使い切られていない場合、使用サイズが表示されます。物理ベースアドレスは、16 進数で与えられます。メモリーバンクが他のバンクによってインタリーブされている場合は、インタリーブファクター(因子)が出力されます。ボード上のメモリーが、非揮発性の変数 `disabled-memory-list` を使ってブート不可になっている場合は、そのことが表示されます。バンクに切り離せないメモリーがあると、固定 (permanent) として出力されます。

`-o disable-at-boot | enable-at-boot`

このオプションは、非揮発性の `disabled-memory-list` 変数を変更することを許可します。これらのオプションは、コマンド要求がない場合は、`-c` オプションの発行や、明白または暗示的なリスティングオプション `-l` と結合して、使用することができます。使用不可メモリーの一覧にあるボード上に構成されているメモリーブロックを無効にするには、`configure` コマンドを `-o enable-at-boot` オプションで使用してください。

`-o extended | normal | quick`

テストレベルを指定するには、`-t` オプションを使用してください。

テストレベル `normal` では、各メモリーセルに 0 と 1 のいずれも格納できることを確認して、さらにすべてのセルが個別にアドレス可能かどうか検査します。テストレベル `quick` では、すべてのメモリーに 0 と 1 を書き込むテストだけ

cfgadm_ac(1M)

を行うため、アドレス線の障害を発見することはできません。extended テストでは、近接したセル同士の干渉の問題をテストするパターンを使用します。デフォルトのテストレベルは、normal です。-t オプションを参照してください。

-o max_errors=#

-t オプションと共に使用して、許可するエラーの最大数を指定します。このオプションを指定しなかった場合は、デフォルト値の 32 が適用されます。

-o timeout=#

構成解除コマンドと共に使用して、自動取り消しによるタイムアウトを設定します。デフォルト値は 60 で、単位は秒です。数値が 0 の場合は、タイムアウトしません。

-t

構成されていないメモリーのバンクをテストします。-o quick | normal | extended オプションでテストレベルを指定してください。

メモリーバンクテストが実行できた場合は、cfgadm コマンドはステータス 0 (成功) で終了します。テストの結果は、接続点のためという条件で利用可能です。

-v

詳細表示オプションです。-t オプションと組み合わせて使用することにより、テストの進行状況と結果を詳細に表示できます。

-x relocate-test

指定されたメモリーバンクで使用されているメモリーの全ページに対し、構成解除コマンドで行われるように、再配置を試みます。この操作の成功は、バンクが構成解除できるかどうかを保証しません。この操作の失敗は、構成解除できなかったことを意味します。このオプションの用途は、テストのみに限定されます。

オペラント 以下のオペラントがサポートされています。

ac#:bank#

メモリーバンクの接続点は、アドレスコントローラー (ac#) ドライバ (ac) のインスタンスによって作成されます。ac ドライバの 1 つのインスタンスは、各システムボードによって作成されますが、CPU/メモリーボードに関連付けられたインスタンスだけは、バンク 0 とバンク 1 の 2 つの接続点を作成します。

このフォームは、cfgadm(1M) によって与えられた論理 ap_id 指定と一致します。これに対応する物理 ap_id のリストは、ファイルの項目にあります。

ac ドライバインスタンスの番号付けは、対応するボードのロット番号とは関係はありません。完全な物理接続点の識別子には、fhc@ に続く、ロット番号の 2 倍を 16 進数で表したロット番号が入ります。

ファイル /devices/fhc@*,f8800000/ac@0,1000000:bank?
接続点

/usr/platform/sun4u/lib/cfgadm/cfgadm_ac.so.1
 ハードウェア固有ライブラリファイル

属性 以下の属性についての説明は、attributes(5)を参照してください。

属性タイプ	属性値
使用条件	SUNWkvm.u

関連項目 cfgadm(1M), cfgadm_sysctrl(1M), config_admin(3CFGADM), attributes(5)

『日本語Solaris7 Sun Enterprise 6X00, 5X00, 4X00, 3X00 システム Dynamic Reconfiguration ユーザーマニュアル』

『特記事項: Sun Enterprise 6x00/5x00/4x00/3x00 システム』

注意事項 EXX00 システムの CPU/メモリーボードの Dynamic Reconfiguration に関する詳細は、『日本語Solaris7 Sun Enterprise 6x00, 5x00, 4x00, 3x00 システム Dynamic Reconfiguration ユーザーマニュアル』を参照してください。

cfgadm_sysctrl(1M)

名前	cfgadm_sysctrl - EXX00 システムボードの管理
形式	<pre>/usr/sbin/cfgadm -c function [-f] [-o disable-at-boot enable-at-boot] [-n -y] sysctrl0:slot# ... /usr/sbin/cfgadm -x quiesce-test sysctrl0:slot# /usr/sbin/cfgadm -x insert-test remove-test sysctrl0:slot# ... /usr/sbin/cfgadm -x set-condition-test=# sysctrl0:slot# ... /usr/sbin/cfgadm [-l] -o disable-at-boot enable-at-boot sysctrl0:slot# ...</pre>
機能説明	<p>sysctrl1 ハードウェア固有ライブラリ、 /usr/platform/sun4u/lib/cfgadm/sysctrl1.so.1 により、E6X00、E5X00、E4X00、E3X00 システム上で、動的再構成を用いてシステムボードの構成や構成解除ができるようになります。これによりシステムが稼働中であっても再起動せずに、I/O ボード、および CPU ボードを、Solaris 用に設定されたシステムのスロットに挿入できるようになります。また同様に、いずれのタイプのボードも稼働中のシステムから切断および取り外しが可能になり、再起動の必要はなくなります。</p> <p>システムスロットはデバイスツリー上で「接続点 (attachment point)」として表現され、システムシャーシ内の各スロットには、この接続点が 1 つずつ設定されます。スロットにボードが設置されていない場合は、受容体の状態は empty と認識されます。ボードの電源が切断されシステムから取り外せる状態の場合は、受容体の状態は disconnected (接続解除) と認識されます。ボードに電源が投入されシステムバスに接続されている状態であれば、受容体の状態は connected (接続) と認識されます。</p> <p>受容体の状態が empty の場合、その受容体の占有装置は unconfigured の状態にあると認識されます。受容体の状態が connected の場合は、その受容体の占有装置は configured または unconfigured のいずれかの状態を取ります。</p> <p>Solaris でボード上のデバイスを利用できるのは、この状態が configured になっている場合です。unconfigured になっている場合、ボード上のデバイスは利用できません。</p> <p>ボードを挿入すると、受容体の状態は empty から disconnected に変わります。ボードを取り外すと、受容体の状態は disconnected から empty に変わります。connected になっているボードを取り外してしまうと、オペレーティングシステムのクラッシュを起こし、システムに修復不可能な障害を与える場合があります。</p>
オプション	<p>オプションについての詳細は、マニュアルページの <code>cfgadm(1M)</code> の項目を参照してください。</p> <p>サポートしているオプションは、以下の通りです。</p> <pre>-c function</pre> <p>状態の変更に使用します。function の部分に connect、disconnect、configure、unconfigure のいずれかを指定します。</p>

configure

占有装置の状態を `configure` に変更します。

受容体の状態が `disconnected` になっている場合は、`configure` 機能はまず受容体への接続を試みます。`connect` 機能の一部として作成される OBP デバイスツリーを調べて Solaris デバイスノードを作成し、要求のあったデバイスを接続します。CPU/メモリーボードの場合は、電源が切断状態にある CPU のリストに、この CPU を加えます。この情報は `psrinfo(1M)` または `psradm(1M)` コマンドを用いて見ることができます。CPU/メモリーボードにはメモリー接続点が 2 つ作成されます。新規に設置したボード上の I/O デバイスを使えるように設定するには、`mount(1M)` および `ifconfig(1M)` コマンドを利用します。新たなプロセッサをオンラインに設定するには、`psradm -n` コマンドを利用します。メモリーバンクのテストおよび設定には `cfgadm_ac(1M)` を利用します。

connect

受容体の状態を `connected` に変更します。

受容体の状態の変更を要求すると、バス信号の接続中はシステムバスを凍結し、その間にボードをテストします。休止処理によってバスは凍結します。この休止処理の間は、すべてのプロセス活動は停止し、すべてのドライバの動作が中断します。この休止処理およびそれに後続する復元処理にはかなりの時間がかかります。またすべてのドライバがこの機能をサポートしているわけではありません。このため、`-x quiesce-test` というオプションを用意しており、これを使用してシステムバスを凍結させている間に、接続されているボードをファームウェアからテストできます。I/O ボードの場合はこの処理には短時間しか費やしません。CPU/メモリーボードの場合は CPU 外部キャッシュのテストが原因で非常に長時間かかります。ここではメモリーのテストはしません。休止処理を開始する前には、実行の確認を促してきます。オプションに `-y` または `-n` を使用すると、この確認作業を省略できます。`disabled-at-boot` に指定されているボードは、接続処理を拒否します。ただし、指定された処理を強制実行する `-f` フラグを利用、または `-o enable-at-boot` を使って指定を無効化した場合は、接続処理が実行されます。`-1` の解説を参照してください。

disconnect

受容体の状態を `disconnected` に変更します。

占有装置の状態が `configure` になっている場合は、まず占有装置を構成解除します。この `disconnect` では休止処理を実行しないので、短時間で終了します。ボードの電源は切断され、取り外しが可能な状態になります。

unconfigure

占有装置の状態を `unconfigureed` に変更します。

この処理の間、ボード上のデバイスは Solaris からは見えない状態になります。I/O ボード上の I/O デバイスは、Solaris のデバイスツリーから削除されます。デバイスが使用中の場合、この `unconfigure` の処理は停止し、使用中であることを報告してきます。この場合はデバイスの使用を停止して、その後で `unconfigure` 処理を再試行する必要があります。CPU/メモリーボードの場合は、ボードの `unconfigure` 処理を実行する前に、メモリーの状態を `unconfigure` に設定しておかなければいけません。ボード上の CPU をオフラインにしてか

cfgadm_sysctrl(1M)

ら、電源を切断して Solaris CPU リストから削除します。接続中のプロセスが存在する CPU をオフラインにすることはできません。CPU のオフライン化についての詳細は、マニュアルページの `psradm(1M)`、`psrinfo(1M)`、`pbind(1M)`、`p_online(2)` の各項目を参照してください。

-f

強制的に、ボードに接続されているブロックの非揮発性変数 `disabled-board-list` を、`disabled-at-boot` に指定します。「特記事項: Sun Enterprise 6x00/5x00/4x00/3x00 システム」のマニュアルも参照してください。

-l

リストを表示させるオプションです。これはマニュアルページの `cfgadm(1M)` で解説しているものと同様の機能です。

`type` フィールドには `cpu/mem`、`mem`、`dual-sbus`、`sbus-upa`、`dual-pci`、`soc+sbus`、`soc+upa`、`disk`、`unknown` のいずれかを指定できます。

ハードウェア固有情報のフィールドは、`[disabled at boot]` `[non-detachable]` `[100 MHz capable]` のように設定されています。

ボードのタイプが `sbus-upa` または `soc+upa` の場合は、まず最初に次のような追加情報が表示されます。

```
[single buffered ffb|double buffered ffb|no ffb installed]
```

ボードのタイプが `disk` の場合は、次のような追加情報が最初に表示されます。

```
{target: # | no disk} {target: # | no disk}
```

-o disable-at-boot | enable-at-boot

非揮発性変数 `disabled-board-list` の変更に使います。この `-o` オプションを使う時は、`-c function` または `-l` オプションと併用します。

`disabled-at-boot` に設定されているボードに関連するブロックを起動時に有効となるようにするには、`-o enable-at-boot` と `-c connect` を併用します。

-x insert-test | remove-test

テストを実行します。

ここで `remove-test` を使うと、ボードを物理的に取り外さなくても、テストシーケンスを自動処理している間、指定したスロットのドライバ状態は `disconnected` から `empty` に変更されます。

また、`insert-test` の方を使った場合は、`remove-test` コマンドで `empty` に指定したスロットのドライバ状態が `isconnected` に変更され、ボードはスロットに挿入されているものとして認識されます。

-x quiesce-test sysctrl0:slot1

テストを実行します。

cfgadm_sysctrl(1M)

このテストの実行により、ボードの接続処理に必要なとなるquiesce処理が実行できる状態になり、現在のソフトウェアおよびハードウェアの構成を保ったままで、システムを確実に休止できるようになります。休止処理できないデバイスやプロセスが存在した場合は、エラーメッセージでその名前を表示します。このコマンドは有効なボード接続点に対してであればどれにでも使用可能ですが、いかなるシステムも必ず slot1 を 1 つ所有しているのので、上に示した形式での使用を推奨しておきます。

`-x set-condition-test=#`
テストを実行します。

状態変更コマンドのポリシーロジックをテストできるように、システムボードの接続点の条件を設定します。新しい設定の内容は、1 から 4 の数字で表現されます。各数字の意味は、以下のようになります。

- 0 unknown (現在の条件が不明)
- 1 ok (障害なし)
- 2 failing (回復可能な障害がある)
- 3 failed (回復不可能な障害がある)
- 4 unusable (接続点が empty)

オペランド 以下のオペランドをサポートしています。

`sysctrl0:slot#` sysctrl ドライバのインスタンス 0 (`sysctrl0`) が、EXX00 システム上のボードに接続点を発行します。接続点の名前には、slot0 から slot15 までの番号を割り当てます。# の部分には 0 から 15 までの数字を入れます。この値がスロット番号を示すこととなります。この形式は、`cfgadm(1M)` を用いた論理的な `ap_id` の指定と一致します。これに対応する物理的な `ap_id` は、次項の【ファイル】に記載してあります。

ファイル `/usr/platform/sun4u/lib/cfgadm/sysctrl.so.1`
ハードウェア固有ライブラリ

`/devices/central@1f,0/fhc@0,f8800000/clock-board@0,900000:slot*`
接続点

属性 属性についての詳細は、マニュアルページの `attributes(5)` を参照してください。

属性タイプ	属性値
使用条件	SUNWkvm.u

関連項目 `cfgadm(1M)`, `cfgadm_ac(1M)`, `ifconfig(1M)`, `mount(1M)`, `pbind(1M)`, `psradm(1M)`, `psrinfo(1M)`, `config_admin(3CFGADM)`, `attributes(5)`

cfgadm_sysctrl(1M)

Sun Enterprise 6x00, 5x00, 4x00 and 3x00 Systems DynamicReconfiguration ユーザーマニュアル,

特記事項:*Sun Enterprise 6x00/5x00/4x00/3x00 Systems*

注意事項 EXX00 システムの CPU/メモリーボードの動的再構成について、更に詳しい情報が必要な場合は、*Sun Enterprise 6x00, 5x00, 4x00 and 3x00 Systems Dynamic Reconfiguration* ユーザーマニュアルを参照してください。

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre> /usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp </pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

closewtmp(1M)

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、リブートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続いて起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のコラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

closewtmp(1M)

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントリングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントリングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

cvcd(1M)

名前	cvcd – 仮想コンソールデーモン						
形式	/platform/platform_name/cvcd						
機能説明	<p>仮想コンソールデーモン cvcd は、一部のプラットフォームで提供されているネットワークコンソールをサポートするサーバープロセスです。cvcd デーモンは、遠隔ホストからのネットワークコンソールの接続を受け付けます (一回につき 1 つのホストのみ)。コンソールの入力はこの接続から読み取られ、cvcredir(7D) を経由して cvc(7D) へ転送されます。</p> <p>同様に、コンソールの出力は cvcredir(7D) から読み取られ、ネットワークコンソールの接続を介して転送されます。cvcd が終了した場合は、コンソールトラフィックは自動的に内部ハードウェアインタフェースによってルート変更されます。</p> <p>通常、cvcd デーモンはシステム起動時に起動されます。各ドメインで同時に実行することができる cvcd プロセスは、1 つだけです。</p> <p>注意: Sun Enterprise 10000 ドメインでは、cvcd は構成ファイル (/etc/ssphostname) を使用して、ネットワークコンソールの接続が許可されるホストの名前を決定します。遠隔コンソールホストの名前が変更された場合は、構成ファイルを編集し、その変更点を反映させる必要があります。</p>						
オペラント	<p>以下のオペラントがサポートされています。</p> <p><i>platform_name</i> パッケージおよびコードで使用されている、Sun の正式なプラットフォーム名。たとえば、Sun Fire 15K サーバーの場合は <i>platform_name</i> は SUNW, Sun-Fire-15000 になります。</p>						
属性	<p>以下の属性については、attributes(5) を参照してください。</p> <table border="1" data-bbox="444 1232 1412 1396"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>アーキテクチャ</td> <td>Sun Enterprise 10000 サーバー、Sun Fire ハイエンドシステム</td> </tr> <tr> <td>使用条件</td> <td>SUNWcvc.u</td> </tr> </tbody> </table>	属性タイプ	属性値	アーキテクチャ	Sun Enterprise 10000 サーバー、Sun Fire ハイエンドシステム	使用条件	SUNWcvc.u
属性タイプ	属性値						
アーキテクチャ	Sun Enterprise 10000 サーバー、Sun Fire ハイエンドシステム						
使用条件	SUNWcvc.u						
関連項目	<p>services(4), attributes(5), cvc(7D), cvcredir(7D)</p> <p>『Sun Enterprise 10000 SSP リファレンスマニュアル』</p> <p>System Management Services (SMS) リファレンスマニュアル</p>						

名前	dd - ファイルの変換とコピー
形式	<code>/usr/bin/dd [operand=value...]</code>
機能説明	<p>dd は、指定した入力ファイルに可能な変換を行なって、指定した出力へコピーします。デフォルトでは、標準入力および標準出力が使用されます。raw 入出力装置の特性を利用するために入出力のブロックサイズで指定することが可能です。サイズは、バイト単位で指定し、数字の後に k、b、または w を付加することができ、それぞれ 1024、512、または 2 の倍数として指定します。また、数字を x で区切るにより乗算を表すことができます。</p> <p>dd は、指定された入力ブロックサイズを用いて、入力データを 1 ブロックずつ読み込みます。その後、実際に渡されたデータブロックを処理します。そのサイズは、指定されたブロックサイズより小さい場合があります。dd は指定された変換処理をブロックに対して行い、結果のデータを、指定された出力ブロックサイズに従ってブロック単位で書き出します。</p> <p>cbs は、ascii、asciib、unblock、ebcdic、ebcdicb、ibm、ibmb、または block 変換が指定されている場合にかぎり使用されます。最初の 2 つの指定では、cbs 文字は変換バッファにコピーされ、任意の指定文字のマッピングが行われます。また後続の空白文字は切り捨てられ、行を送信する前に復帰改行が追加されます。残りの 3 つの指定では、復帰改行までの文字が変換バッファに読み込まれ、サイズ cbs の出力レコードを構成するために空白文字が追加されます。ASCII ファイルは復帰改行文字を含むものとして、cbs が指定されていないかまたは 0 であると、ascii、asciib、ebcdic、ebcdicb、ibm、ibmb の各オプションは入力ファイルのブロック構造を変更せずに文字セットを変換します。unblock および block の各オプションは単純なファイルコピーを行います。</p> <p>終了後に、dd は全体および部分的な入出力ブロック数を報告します。</p>
オペランド	<p>以下のオペランドが指定できます。</p> <p><code>if=file</code> 入力パス名。デフォルトは標準入力です。</p> <p><code>of=file</code> 出力パス名。デフォルトは標準出力です。seek=expr 変換が指定されていないとき、conv=notrunc も指定されていないければ、コピー処理の実行前に出力ファイルは切り捨てられます。seek=expr が指定され、conv=notrunc が指定されていないければ、コピー処理の結果としては dd がシークする出力ファイル中のブロックは保持されますが、出力ファイルのその他の部分は保持されません。シークするサイズと入力ファイルのサイズの合計が出力ファイルの元のサイズより小さい場合、コピー処理により出力ファイルは小さくなります。</p> <p><code>ibs=n</code> 入力ブロックサイズを n バイトとします。デフォルト値は 512 です。</p> <p><code>obs=n</code> 出力ブロックサイズを n バイトとします。デフォルト値は 512 です。</p>

dd(1M)

bs=*n*

入力ブロックサイズと出力ブロックサイズをともに *n* バイトとします。この指定は **ibs=** と **obs=** 指定よりも優先されます。**sync**、**noerror**、**notrunc** 以外の変換が1つも指定されない場合、各入力ブロックは複数の短いブロックを1つにまとめる処理は行われず、それぞれ単独のブロックとして出力側にコピーされます。

cbs=*n*

block と **unblock** 用の変換ブロックサイズを *n* バイトに指定します。デフォルト値は 0 です。**cbs=** を指定しないかまたは 0 を指定した場合、**block** や **unblock** を使うと結果は予測できません。

このオプションは ASCII または EBCDIC 変換を指定した場合にだけ有効です。**ascii** と **asciib** オペランドを指定した場合、後続の空白文字を消去する前に文字が ASCII に変換されるという点を除き、入力処理は **unblock** オペランド指定時と同じです。**ebcdic**、**ebcdicb**、**ibm**、または **ibmb** オペランドを指定した場合、後方に空白文字を追加した後で文字が EBCDIC または IBM EBCDIC に変換されるという点を除き、入力処理は **block** オペランド指定時と同じです。

files=*n*

終了する前に、*n* 個の入力ファイルをコピーして連結します (入力が磁気テープまたは同様な装置の場合だけ有効です)。

skip=*n*

コピーを開始する前に、指定された入力ブロックサイズを用いて *n* 個の入力ブロックを読み飛ばします。シーク可能なファイルに対しては、システムはそれらのブロックを読み込むか、あるいはシークを行います。シーク不可能なファイルに対しては、ブロックを読み込んで、そのデータを捨てます。

iseek=*n*

コピーを行う前に、入力ファイルの先頭から *n* 個のブロックをシークします (**skip** の動作が遅いディスクファイルに適しています)。

oseek=*n*

コピーを行う前に、出力ファイルの先頭から *n* 個のブロックをシークします。

seek=*n*

コピーを行う前に、出力ファイルの先頭から *n* 個のブロックをスキップします (指定された出力ブロックサイズを使用)。シーク不可能なファイルに対しては、既存のブロックを読み込み、現在のファイルの終わり位置から指定されたオフセット位置までの間に空白があれば、その空白を NULL バイトで埋めます。シーク可能なファイルに対しては、指定されたオフセット位置までをシークするか、またはシーク不可能なファイルの場合と同様にブロックを読み込みます。

count=*n*

n 個の入力ブロックだけをコピーします。

conv=*value*[, *value*...]

1 つ以上の *value* をコンマで区切って記述します。各 *value* は以下のいずれかです。

ascii EBCDIC を ASCII に変換します。

asciib BSD 互換の文字変換を使用して、EBCDIC を ASCII に変換します。

<code>ebcdic</code>	ASCII を EBCDIC に変換します。復帰改行のない固定長の ASCII レコードを変換する場合は、前もって <code>dd conv=unblock</code> でパイプラインを整えてください。
<code>ebcdicb</code>	BSD 互換の文字変換を使用して、ASCII を EBCDIC に変換します。復帰改行のない固定長の ASCII レコードを変換する場合は、前もって <code>dd conv=unblock</code> でパイプラインを整えてください。
<code>ibm</code>	ASCII から EBCDIC への変換とわずかに異なるマップを使用します。復帰改行のない固定長の ASCII レコードを変換する場合は、前もって <code>dd conv=unblock</code> でパイプラインを整えてください。
<code>ibmb</code>	BSD 互換の文字変換を使用して ASCII から EBCDIC への変換とわずかに異なるマップを使用します。復帰改行のない固定長の ASCII レコードを変換する場合は、前もって <code>dd conv=unblock</code> でパイプラインを整えてください。
<code>ascii</code> (または <code>asciib</code>)、 <code>ebcdic</code> (または <code>ebcdicb</code>)、 <code>ibm</code> (または <code>ibmb</code>) のうち 2 つ以上を同時に指定することはできません。	
<code>block</code>	入力データを、入力ブロック境界に関係なく、復帰改行文字もしくは EOF で終わる可変長レコードの集まりと見なします。各レコードは、変換ブロックサイズで指定した長さの固定長レコードに変換されます。入力行に復帰改行文字があれば削除されます。変換ブロックサイズより短い行に関しては、ブロックを埋めるために空白文字が付加されます。変換ブロックサイズより長い行に関しては、ブロックサイズを満たす最大長になるように文字が捨てられます。切り捨てられた行の総数が報告されます。
<code>unblock</code>	固定長のレコードを可変長に変換します。変換ブロックサイズと等しい長さ分のバイト (それより短いデータしか残っていないければそのすべて) を読み込み、後続の空白文字を削除して、復帰改行文字を付加します。
<code>block</code> と <code>unblock</code> を同時に指定することはできません。	
<code>lcase</code>	<code>LC_CTYPE</code> カテゴリ中のキーワードの <code>tolower</code> で指定された大文字を、対応する小文字にマップします。マッピングを指定されなかった文字は、この変換によって変更されることはありません。
<code>ucase</code>	<code>LC_CTYPE</code> カテゴリ中のキーワードの <code>toupper</code> で指定された小文字を、対応する大文字にマップします。マッピングを指定されなかった文字は、この変換によって変更されることはありません。
<code>lcase</code> と <code>ucase</code> を同時に指定することはできません。	

dd(1M)

swab	入力データを2バイトずつの対になっているものとし、各々の対についてバイトの値を交換します。入力レコードの長さが奇数バイトの場合には、最終バイトは無視されます。
noerror	入力エラーが発生しても処理を停止しません。入力エラーが起これば、標準エラー出力に診断メッセージと、入力および出力ブロック数が出力されます。ブロック数の形式は、正常に終了したときに出力されるものと同じです。sync 変換が指定されたときは、入力データのうち得られなかったバイトは NULL バイトに置き換えられて通常どおりに処理されます。sync が指定されなければ、入力ブロックは出力上には現れません。
notrunc	出力ファイルを切り捨てません。今回の dd 呼び出しで明示的に書き出されなかったブロックも出力ファイル中に保持します(前述の of=file オペランドの説明を参照)。
sync	入力ブロックに NULL バイトを付加して、サイズが ibs= で指定した値に等しくなるようにします。なお block または unblock も指定されている場合には、NULL バイトの代わりに空白文字を付加します。

conv= 以外のオペランドが複数回指定されたときは、最後に記述された operand=value が有効となります。

bs=、cbs=、ibs=、obs= に関しては、バイト単位でサイズを指定する式をアプリケーション側で提供しなければなりません。式 *expr* は以下のいずれかです。

1. 正の 10 進数。
2. 正の 10 進数の後に 1024 倍を示す k を付加したもの。
3. 正の 10 進数の後に 512 倍を示す b を付加したもの。
4. 2 つ以上の正の 10 進数 (k や b を付加してもなくてもよい) を文字 x で区切ったもの。その 2 つの値を乗算した結果を表す。

オペランドはすべて、入力データを読み込む前に処理されます。

使用法 ファイルが 2G バイト (2³¹ バイト) 以上ある場合の dd の動作については、largefile(5) を参照してください。

使用例 例 1 テープドライブ 0 から 1 へのコピー

次のコマンドは、標準的なデバイス命名規約を用いて、テープドライブ 0 から 1 へコピーします。

```
example% dd if=/dev/rmt/0h of=/dev/rmt/1h
```

例 2 標準入力の先頭の 10 バイトの削除

次のコマンドは、標準入力の先頭の 10 バイトを削除します。

```
example% dd ibs=10 skip=1
```

例 2 標準入力の先頭の 10 バイトの削除 (続き)

例 3 テープを ASCII ファイルに読み込む

次のコマンドは、テープブロックごとに 10 個の 80 バイト EBCDIC カードイメージにブロック化した EBCDIC テープを ASCII ファイル `x` に読み込みます。

```
example% dd if=/dev/tape of=x ibs=800 cbs=80 conv=ascii,lcase
```

例 4 テープの書き込みに `conv=sync` を使用

次のコマンドは、テープに書き込む場合に `conv=sync` を使用する例です。

```
example% tar cvf - . | compress | dd obs=1024k of=/dev/rmt/0 conv=sync
```

環境 dd の実行に影響を与える環境変数 `LC_CTYPE`、`LC_MESSAGES`、`NLSPATH` についての詳細は、`environ(5)` を参照してください。

終了ステータス 以下の終了ステータスが返されます。

0 ファイルは正常にコピーされた

>0 エラーが発生した

`noerror` 変換が指定されていないときに入力エラーが発生すると、部分的に生成された出力ブロックがあればそれを出力ファイルに書き出し、診断メッセージを出力し、コピー処理を中止します。その他のエラーを検出した場合には、診断メッセージを出力し、コピー処理を中止します。

属性 次の属性については `attributes(5)` のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWcsu

関連項目 `cp(1)`、`sed(1)`、`tr(1)`、`attributes(5)`、`environ(5)`、`largefile(5)`

診断 `f+p records in(out)` 読み取られた (書き込まれた) 完全 (f) および部分 (p) ブロック数

注意事項 ブロックサイズが異なるファイルシステム間でファイルのコピーを行う場合には、`dd` を使用しないでください。

ファイルをコピーするのにブロック型デバイスを使用すると、最終ブロックをブロック境界にあわせるために余分な NULL バイトがファイルに追加されます。

`dd` が `ibs=X` および `obs=Y` オペランドを用いてパイプから読み取ると、その出力はつねにサイズ `Y` の固まりでブロック化されます。`bs=z` が使用されると、出力ブロックはその時点でパイプから読み取ることができる大きさになります。

dd(1M)

dd を使用してテープデバイスにファイルをコピーする場合、ファイルのサイズはデバイスのセクタサイズ (たとえば 512K) の倍数でなければなりません。任意のサイズのファイルをテープデバイスにコピーする場合は、tar(1) または cpio(1) を使用してください。

SIGINT が発生した場合、dd はステータス情報を標準エラー出力に書き出して処理を終了します。その他のシグナル発生時には、標準的な動作を行います。

名前	df - 使用可能なディスクブロックおよびファイル数の表示
形式	<pre> /usr/bin/df [-F FSType] [-abegklntV] [-o FSType-specific_options] [block_device directory file resource...] /usr/xpg4/bin/df [-F FSType] [-abegklntPtV] [-o FSType-specific_options] [block_device directory file resource...] </pre>
機能説明	<p>df コマンドは、マウントされているもしくはマウントされていないファイルシステムが使用しているディスク容量を表示します。さらに、使用中の容量、使用可能な容量、ファイルシステムの全容量のうちどの程度が使用されたかを表示します。対象のファイルシステムとしては、デバイス、または特定のファイルシステム上のファイルあるいはディレクトリを指定します。</p> <p>オペランドもオプションも指定せずに df を実行すると、全ファイルシステムに関する情報が出力されます。</p> <p>df はすべての <i>FSTypes</i> (ファイルシステムタイプ) をサポートしません。</p>
オプション	<p>以下のオプションは、<code>/usr/bin/df</code> と <code>/usr/xpg4/bin/df</code> で指定できます。</p> <p>-a /etc/mnttab のエントリ (mnttab(4) を参照) に ignore オプションセットを含む ファイルシステムについても情報を出力します。</p> <p>-b 使用可能な容量の合計を、K バイト単位で報告します。</p> <p>-e 使用可能なファイル数だけを報告します。</p> <p>-F <i>FSType</i> df コマンドが動作する <i>FSType</i> を指定します。-F オプションはマウントされていないファイルシステムを使用するためのオプションです。<i>FSType</i> は、このオプションで指定するか、あるいは <code>/etc/vfstab</code> 内の記述 (vfstab(4) を参照) から特定できるようにしておく必要があります。後者の場合、具体的には <i>directory</i>、<i>block_device</i>、または <i>resource</i> 引数の指定値とテーブル中のエントリの値を一致させるか、あるいは <code>/etc/default/fs</code> を参照して決定されます。詳細は <code>default_fs(4)</code> を参照してください。</p> <p>-g <code>statvfs(2)</code> 構造体全体を報告します。このオプションはマウントされているファイルシステムに対してのみ有効であり、<code>-o</code> オプションと同時に指定することはできません。このオプションは、<code>-b</code>、<code>-e</code>、<code>-k</code>、<code>-n</code>、<code>-P</code>、<code>-t</code> の各オプション (指定されている場合) よりも優先されます。</p> <p>-k 割り当てられているディスク容量を K バイト単位で出力します。1 つのファイルシステムにつき 1 行の情報が出力されます。情報の内容は、ファイルシステム名、そのファイルシステムに割り当てられている容量の合計、既存のファイルに割り当てられている容量の合計、特権を持たないユーザーが新たなファイルを生成する場合に使用できる容量の合計、そのファイルシステム上の全ファイルに現在割り当てられている通常使用可能な容量の割合</p>

df(1M)

	(パーセント単位)です。このオプションは、 <code>-b</code> 、 <code>-e</code> 、 <code>-n</code> 、 <code>-t</code> の各オプション (指定されている場合) よりも優先されます。
<code>-l</code>	ローカルファイルシステムについての情報だけを報告します。このオプションはマウントされているファイルシステムに対してのみ有効であり、 <code>-o</code> オプションと同時に指定することはできません。
<code>-n</code>	<i>FSType</i> の名前だけを報告します。オペラント指定を省略すると、このオプションはマウントされているファイルシステムタイプの一覧を出力します。このオプションはマウントされているファイルシステムに対してのみ有効であり、 <code>-o</code> オプションと同時に指定することはできません。
<code>-o FSType-specific_options</code>	<i>FSType</i> 固有のオプションを指定します。オプションとオプションの間はコマンドだけで区切り、空白は入れないでください。詳細については <i>FSType</i> コマンド用のマニュアルページを参照してください。
<code>-t</code>	合計値を含む完全なリストを出力します。このオプションは、 <code>-b</code> 、 <code>-e</code> 、 <code>-n</code> の各オプション (指定されている場合) よりも優先されます。
<code>-v</code>	指定されたコマンド行の全内容のエコーだけを行い、コマンド自体は実行しません。コマンド行の内容としては、ユーザーが指定したオプションやオペラントに加え、 <code>/etc/mnttab</code> 、 <code>/etc/vfstab</code> 、 <code>/etc/default/fs</code> の各ファイルから得られた情報が付加されます。このオプションは、コマンド行の記述内容の正当性を検査する目的に使用できます。
<code>/usr/xpg4/bin/df</code>	以下のオプションは、 <code>/usr/xpg4/bin/df</code> でのみ指定できます。
<code>-P</code>	<code>-k</code> オプションと同じですが、出力の単位は 512 バイトとなります。
オペラント	<code>df</code> は、 <i>block_device</i> 、 <i>directory</i> 、 <i>file</i> の優先度に従ってオペラントを解釈します。以下のオペラントを指定できます。
<i>block_device</i>	ブロック型特殊デバイス (たとえば <code>/dev/dsk/c1d0s7</code>) を指定します。対応するファイルシステムはマウントされている必要はありません。
<i>directory</i>	既存のディレクトリ名を指定します。
<i>file</i>	有効なファイル名を指定します。 <code>df</code> は <i>file</i> があるファイルシステムについて報告します。
<i>resource</i>	NFS リソース名を指定します。
使用法	ファイルが 2 ギガバイト (2^{31} バイト) 以上ある場合の <code>df</code> の動作については、 <code>largefile(5)</code> を参照してください。

使用例	<p>例 1 df コマンドを使用して、ファイルシステム /usr に関する情報を出力する</p> <p>次の例は、ファイルシステム /usr に関する情報を出力します。</p> <pre>example% /usr/xpg4/bin/df -P /usr</pre> <p>例 2 df コマンドを使用して、ファイルシステム /usr に関する情報を出力する (/usr/src が /usr ファイルシステムの一部である場合)</p> <p>/usr/src が /usr ファイルシステムの一部である場合、次の例も上記の例と同じ結果を出力します。</p> <pre>example% /usr/xpg4/bin/df -P /usr/src</pre> <p>例 3 df コマンドを使用して、すべての ufs ファイルシステムの i ノード使用率を表示する</p> <p>次の例は、すべての ufs ファイルシステムにおける i ノードの使用率を表示します。</p> <pre>example% /usr/bin/df -F ufs -o i</pre>
環境	<p>SYSV3 この環境変数はデフォルトの df の動作を無効にして、INTERACTIVE UNIX システムと SCO UNIX のインストールスクリプトとの互換性を提供するために使用します。SYSV3 は互換性だけを目的とした環境変数なので、新しいスクリプトでは使用しないでください。</p> <p>設定した場合、通常、「ファイル」を表示するヘッダーはすべて、「ノード」を表示するようになります。df の実行に影響を与える環境変数 LC_CTYPE、LC_MESSAGES、NLSPATH についての詳細は、environ(5) を参照してください。</p>
終了ステータス	<p>以下の終了ステータスが返されます。</p> <p>0 正常終了</p> <p>>0 エラーが発生した</p>
ファイル	<p>/dev/dsk/* ディスクデバイス</p> <p>/etc/default/fs ローカルファイルシステムタイプのデフォルト値。デフォルト値は /etc/default/fs の以下のフラグに対して設定することができる</p> <p> 例: LOCAL=ufs</p> <p> LOCAL: FSType が指定されなければ、コマンドのデフォルトパーティションになる</p> <p>/etc/mnttab マウントテーブル</p> <p>/etc/vfstab 各ファイルシステム用のデフォルトパラメタ</p>
属性	<p>次の属性については attributes(5) のマニュアルページを参照してください。</p>

df(1M)

/usr/bin/df

属性タイプ	属性値
使用条件	SUNWcsu

/usr/xpg4/bin/df

属性タイプ	属性値
使用条件	SUNWxcu4

関連項目 find(1), df_ufs(1M), mount(1M), statvfs(2), default_fs(4), mnttab(4),
vfstab(4), attributes(5), environ(5), largefile(5), XPG4(5)

注意事項 ファイルシステムで UFS ログが有効になっている場合、ログで使用されるディスク容量は df の報告に反映されます。ログは、ファイルシステムの空きブロックから割り当てられ、ファイルシステム 1G バイト当り 約 1M バイトから最大約 64M バイトのサイズになります。

名前	du - ディスク使用状況の要約
形式	<code>/usr/bin/du [-adkLr] [-s -o] [file...]</code> <code>/usr/xpg4/bin/du [-a -s] [-krx] [file...]</code>
機能説明	<p>du ユーティリティは、指定されたファイルを根とするディレクトリ階層に関して、割り当てたファイル領域のサイズと、各サブディレクトリに割り当てたファイル領域のサイズを標準出力に書き出します。ディレクトリ型のファイルに割り当てた領域のサイズは、そのディレクトリ自体に割り当てた領域だけでなく、そのディレクトリから派生しているファイル階層構造上のすべてのファイルに割り当てた領域の合計サイズです。</p> <p>複数のリンクを持つファイルは、1 個のエントリとして数えられ出力されます。レポート中で選択されるディレクトリエントリは決められていません。ファイルのサイズは、デフォルトでは 512 バイト単位で出力されます。端数があれば切り上げられます。</p>
<code>/usr/xpg4/bin/du</code>	du がファイル属性を認識できない場合、およびディレクトリに対して読み取りができない場合 (stat(2) 参照)、エラー状態を報告し、最終的な終了ステータスにそれが反映されます。
オプション	<p>以下のオプションは、<code>/usr/bin/du</code> と <code>/usr/xpg4/bin/du</code> の両方で指定できます。</p> <ul style="list-style-type: none"> -a デフォルトの出力内容に加え、指定したファイルを根とするディレクトリ階層に含まれるディレクトリ以外の各ファイルのサイズを書き出します。このオプションを指定しても省略しても、<i>file</i> オペランドで指定されたディレクトリ以外のファイルに関してのサイズ情報は必ず出力されます。 -k ファイルのサイズを、デフォルトの 512 バイト単位ではなく 1024 バイト単位で出力します。 -s デフォルトの出力ではなく、指定した各ファイルの合計サイズだけを報告します。
<code>/usr/bin/du</code>	<p>以下のオプションは、<code>/usr/bin/du</code> でのみ指定できます。</p> <ul style="list-style-type: none"> -d ファイルシステムの境界をまたいだ処理は行いません。たとえば “<code>du -d /</code>” に対しては、ルートパーティションに関してだけ出力を行います。 -L シンボリックリンクの処理に、リンク自体ではなく、シンボリックリンクが参照しているファイルまたはディレクトリを使用します。 -o 親ディレクトリのディスク使用量の合計に、子ディレクトリのディスク使用量を含めません。このオプションを省略した場合、ディレクトリのディスク使用量は、指定したディレクトリ中のファイルのディスク使用量に、そのディレクトリの下にあるすべてのディレクトリ中のファイルのディスク使用量を加えたものになります。-s オプションを同時に指定した場合は何もしません。 -r 読み取れないディレクトリや、オープンできないファイルなどに対してメッセージを出力します。デフォルトでは出力しません。

du(1M)

/usr/xpg4/bin/du	<p>以下のオプションは、<code>/usr/xpg4/bin/du</code> でのみ指定できます。</p> <p><code>-r</code> デフォルトで、読み取れないディレクトリや、オープンできないファイルなどに対してメッセージを出力します。</p> <p><code>-x</code> ファイルのサイズを評価する際に、<code>file</code> オペランドで示したファイルと同一のデバイスのファイルだけを対象とします。</p> <p>オペランド 以下のオペランドを指定できます。</p> <p><code>file</code> サイズを調べる対象となるファイルのパス名。<code>file</code> を1つも指定しないと、現在のディレクトリが用いられる</p> <p>出力 <code>du</code> の出力は、ファイルに割り当てた領域のサイズとファイル名で構成されます。</p> <p>使用法 ファイルが 2G バイト (2^{31} バイト) 以上ある場合の <code>du</code> の動作については、<code>largefile(5)</code> を参照してください。</p> <p>環境 <code>du</code> の実行に影響を与える環境変数 <code>LC_CTYPE</code>、<code>LC_MESSAGES</code>、<code>NLSPATH</code> についての詳細は、<code>environ(5)</code> を参照してください。</p> <p>終了ステータス 以下の終了ステータスが返されます。</p> <p>0 正常終了</p> <p>>0 エラーが発生した</p> <p>属性 次の属性については <code>attributes(5)</code> のマニュアルページを参照してください。</p>						
/usr/bin/du	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">属性タイプ</th> <th style="text-align: center;">属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWcsu</td> </tr> <tr> <td>CSI</td> <td>対応済み</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWcsu	CSI	対応済み
属性タイプ	属性値						
使用条件	SUNWcsu						
CSI	対応済み						
/usr/xpg4/bin/du	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">属性タイプ</th> <th style="text-align: center;">属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWxcu4</td> </tr> <tr> <td>CSI</td> <td>対応済み</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWxcu4	CSI	対応済み
属性タイプ	属性値						
使用条件	SUNWxcu4						
CSI	対応済み						
関連項目	<p><code>ls(1)</code>, <code>stat(2)</code>, <code>attributes(5)</code>, <code>environ(5)</code>, <code>largefile(5)</code>, <code>XPG4(5)</code></p> <p><i>Solaris</i> のシステム管理 (第 1 巻)</p>						
注意事項	<p>複数のリンクを持つファイルは、1 回だけ数えられます。ただし、異なったディレクトリ中のファイル間にリンクが存在し、それらのディレクトリが異なるファイルシステムに存在している場合、<code>du</code> はファイルを重複して数えます。</p> <p>ホールを含むファイルの場合、ブロック数として 誤った値を出力することがあります。</p>						

名前	ffbcfig, SUNWffb_config – FFB グラフィックスアクセラレータの設定
形式	<pre> /usr/sbin/ffbcfig [-dev <i>device-filename</i>] [-res <i>video-mode</i> [now try] [noconfirm nocheck]] [-file machine system] [-deflinear true false] [-defoverlay true false] [-linearorder first last] [-overlayorder first last] [-expvis enable disable] [-sov enable disable] [-maxwids <i>n</i>] [-extovl enable disable] [-g <i>gamma-correction-value</i>] [-gfile <i>gamma-correction-file</i>] [-propt] [-prconf] [-defaults] /usr/sbin/ffbcfig [-propt] [-prconf] /usr/sbin/ffbcfig [-help] [-res ?] </pre>
機能説明	<p>ffbcfigは、FFB グラフィックスアクセラレータおよび FFB 対応の X11 ウィンドウシステムのデフォルトの一部を設定します。</p> <p>ffbcfigの1番目の形式では、指定したオプションを OWconfigファイルに保存します。これらのオプションは、次にウィンドウシステムをそのデバイスで実行するときに FFB デバイスを初期化するために使用されます。OWconfig ファイル内のオプションの更新は、異なるウィンドウセッションや再起動したシステムでも有効となります。</p> <p>-prconf、-propt、-help、-res? オプションだけを起動する2番目と3番目の形式では、OWconfig ファイルは更新されません。また、3番目の形式では、その他のオプションはすべて無視されます。</p> <p>オプションは、一度に1つの FFB デバイスに対してのみ指定することができます。複数の FFB デバイスに対してオプションを指定するには、ffbcfig を複数回起動する必要があります。</p> <p>ffbcfig で指定できるのは、FFB 固有のオプションだけです。デフォルトの表示色数、デフォルトの画像表示形式クラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください。詳細については、OpenWindows デスクトップ・リファレンスマニュアルを参照してください。</p> <p>ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは、/etc/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/usr/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。</p> <p>これらの標準 OWconfig ファイルのどちらもスーパーユーザーのみが書き込みを行います。したがって、スーパーユーザーが所有する ffbcfig プログラムは、setuid による root の権限で実行されます。</p>
オプション	<pre> -dev <i>device-filename</i> FFB 特殊ファイルを指定します。デフォルトは/dev/fbs/ffb0 です。 -file <i>machine</i> <i>system</i> 更新する OWconfig ファイルを指定します。machine を指定すると、 /etc/openwin ディレクトリツリーにあるマシン固有の OWconfig ファイルが使 </pre>

fbconfig(1M)

用されます。system を指定すると、/usr/openwin ディレクトリツリーにある共通のOWconfig ファイルが使用されます。ファイルがない場合は、新たに生成されます。

`-res video-mode [now | try [noconfirm | nocheck]]`

指定した FFB デバイスに接続されているモニターを制御する際に使われる表示モードを指定します。

表示モードの形式は `widthxheightxrate` で、`width` はピクセル単位の画面幅、`height` はピクセル単位の画面の高さ、`rate` は画面を垂直方向に再描画する周期です。

960x680x112s や 960x680x108s の s 接尾辞は、これらが立体表示モードであることを意味します。640x480x60i や 768x575x50i の i 接尾辞は、インタレース表示タイミングを有効にします。この接尾辞がない場合は、ノンインタレースタイミングが使用されます。

`-res` (「形式」に記されている 3 番目の形式) にリフレッシュレートを指定する際は、値の直前に x の代わりに @ を使用することができます。たとえば、1280x1024@76 のように指定することができます。

一部の表示モードは、FFB の一部のバージョンのみが対応しています。また、FFB が対応している表示モードには、モニターが対応していないものもあります。FFB デバイスとモニターの両方が対応している表示モードのリストは、`-res ? オプション付きの fbconfig` を実行することによって得ることができます。

FFB が対応している表示モードのリストを以下に示します。

表示モード	説明
1024x768x60	
1024x768x70	
1024x768x75	
1024x768x77	
1024x800x84	
1152x900x66	
1152x900x76	
1280x800x76	
1280x1024x60	
1280x1024x67	
1280x1024x76	
960x680x112s	(立体表示)

表示モード	説明
960x680x108s	(立体表示)
640x480x60	
640x480x60i	(インタレース)
768x575x50i	(インタレース)
1440x900x76	(高解像度)
1600x1000x66	(高解像度)
1600x1000x76i	(高解像度)
1600x1280x76	(高解像度)
1920x1080x72	(高解像度)
1920x1200x70	(高解像度)

記号名

便宜上、表示モードのいくつかには記号名が定義されています。*widthxheightxrate* の形式の代わりに、記号名を *-res* の引数として指定することができます。記号名 *none* は、ウィンドウシステムを実行すると、画面の解像度は現在デバイスにプログラムされている表示モードになることを意味します。

記号名	対応する表示モード
<i>svga</i>	1024x768x60
<i>1152</i>	1152x900x76
<i>1280</i>	1280x1024x76
<i>stereo</i>	960x680x112s
<i>ntsc</i>	640x480x60i
<i>pal</i>	768x575x50i
<i>none</i>	(デバイスでプログラムされている表示モード)

-res オプションには、表示モードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定することができます。

now

OWconfig ファイルの表示モードを更新するとともに、FFB デバイスが指定した表示モードにただちにプログラムされます (この機能は、ウィンドウシステムを開始する前に表示モードを変更する際に便利です)。

ffbconfig(1M)

対象となるデバイスが稼働している間 (たとえば、ウィンドウシステムの稼働中) に、この追加オプションを `ffbconfig` に指定することはお勧めしません。予期しない結果になることもあります。 `now` オプションを指定して `ffbconfig` コマンドを実行する場合は、最初にウィンドウシステムを終了してください。 `now` オプションがウィンドウシステムのセッション中に使用された場合、表示モードはただちに変更されますが、画面の幅や高さはそのセッションが終了して次のセッションに入るまで変更されません。さらに、立体表示モードではシステムが変更を認識しないことがあります。したがって、ウィンドウシステムの稼働中には絶対に `now` オプションを指定しないでください。

`noconfirm`

確認と警告メッセージを省略し、要求された表示モードにプログラムします。

`-res` オプションを指定した際に、システムが使用可能であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために `ffbconfig` のデフォルトの動作では、この問題についての警告メッセージと処理を継続するかどうかを確認するメッセージを表示します。このオプションは、`ffbconfig` がシェルスクリプトから実行されている場合に便利です。

`nocheck`

モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定された表示モードは、現在接続されているモニターに適切かどうかにかかわらず受け付けられます。このオプションは、FFB デバイスに異なるモニターを接続する場合に便利です。このオプションを指定すると、`noconfirm` も指定されます。

`try`

指定した表示モードに試験的にプログラムされます。ユーザーは、指定した表示モードを使用する場合は、メッセージが表示されてから 10 秒以内に `y` と入力します。表示されたモードを使用しない場合は、10 秒以内に任意の文字を入力します。 `y` または `Return` キー以外の文字の入力は、すべて「使用しない」とみなされ、以前の表示モードに戻され、`OWconfig` ファイル中の表示モードは書き換えられません。その他の指定されたオプションは有効となります。 `Return` キーの入力があった場合は、新しい表示モードを保持するかどうかを `yes` または `no` で確認するメッセージが表示されます。このオプションを指定すると、`now` サブオプションも指定されます。(`now` サブオプションの注意を参照)。

`-deflinear true | false`

FFB には、2 種類の画像表示形式があります。リニア画像と非リニア画像です。リニア画像はガンマ補正され、非リニア画像は補正されません。リニア画像版も非リニア画像版も、ともに持つ画像表示形式が 2 つあります。24 ビット `TrueColor` と 8 ビット `StaticGray` です。

`-deflinear true` を指定すると、デフォルトの画像表示形式として、デフォルトで選択されたオプション (特に、Xsun(1)の `defdepth` および `defclass` オプション。詳細は *OpenWindows* デスクトップ・リファレンスマニュアルのマニュアルページを参照) を満たすリニア画像を設定します。具体的には、デフォルトの画像表示形式の選択オプションは、Xsun(1)の `defdepth` および `defclass` オプションによって設定されたものです。

`-deflinear false` を指定すると、または他のデフォルトで選択されたオプションを満たすリニア画像がない場合は、これらの他のオプションを満たす非リニア画像がデフォルトとして選択されます。

FFB にはリニアオーバーレイ画像表示形式がないため、オプションが存在する場合は、このオプションを使用することはできません。

`-defoverlay true | false`

FFB が、残りの FFB 画像から切り離されたピクセルを持つ 8 ビット疑似カラー画像を提供します。これを、オーバーレイ画像といいます。この画像表示形式で作成されたウィンドウは、他の画像表示形式で作成されたウィンドウに影響を与えません。逆に、他の画像表示形式で作成されたウィンドウは、オーバーレイウィンドウに影響を与えます。この画像表示形式では、256 種類の不透明カラーの値による拡張オーバーレイを使用することができます。-maxwid を参照してください。

`-defoverlay` に `true` を指定すると、オーバーレイ画像がデフォルト画像になります。`-defoverlay` に `false` を指定すると、他のデフォルトで選択された `defdepth` および `defclass` オプションを満たすオーバーレイでない画像表示形式が、デフォルトの画像表示形式として選択されます。詳細は、*OpenWindows* デスクトップ・リファレンスマニュアルのマニュアルページを参照してください。

`-defoverlay true` を使用する場合は、`openwin` コマンド行で選択されたデフォルトの深さとクラスは常に 8 ビット疑似カラーである必要があります。それ以外の場合は、警告メッセージが出力され、`-defoverlay` オプションは `false` として扱われます。`-deflinear` オプションが存在する場合、FFB にはリニアオーバーレイ画像表示形式がないため、このオプションは使用することができません。

`-linearorder first | last`

`first` を指定すると、FFB 画面用の X11 画面画像表示形式リスト上で、リニア画像が非リニア画像より前に表示されます。`last` を指定すると、非リニア画像は、リニア画像より前に表示されます。

`-overlayorder first | last`

`first` を指定すると、FFB 画面用の X11 画面画像表示形式リスト上で、8 ビット疑似カラーオーバーレイ画像が、非オーバーレイ画像より前に表示されます。`last` を指定すると、非オーバーレイ画像は、オーバーレイ画像より前に表示されます。

`-expvis enable | disable`

`enable` を指定すると、OpenGL Visual Expansion が起動されます。選択された画像表示形式グループ (8 ビット `PseudoColor`、24 ビット `TrueColor` など) は、画面画像表示形式リストで見つけることができます。

ffbconfig(1M)

-sov enable | disable

enable を指定すると、ルートウィンドウの `SERVER_OVERLAY_VISUALS` 属性が有効になります。SOV 画像が転送され、それらの透過タイプ、値、階層は、この属性によって参照することができます。disable を指定すると、`SERVER_OVERLAY_VISUALS` 属性は定義されません。SOV 画像は転送されません。

-maxwids *n*

ウィンドウ ID (WID) として使用するために予約される最大数の FFB チャンネルピクセル値を指定します。オーバーレイカラーマップのピクセル値の残りは、通常の X11 の未使用のカラーピクセルのために使用されます。確保された WID は、(XGL などの) 3次元グラフィックスウィンドウ、MBX ウィンドウと、デフォルト以外の画像表示形式をもつウィンドウにより発生順に割り当てられます。X チャンネルコードの 0 から (255-*n*) は、未使用のカラーピクセルです。(255-*n*+1) から 255 の X チャンネルコードは、WID として使用するために予約されます。FFB と FFB2 の有効な値は、1、2、4、8、16、32 です。FFB2+ の有効な値は、1、2、4、8、16、32、64 です。

-extovl enable | disable

このオプションは FFB2+ のみで使用することができます。enable を指定すると、拡張オーバーレイを使用することができます。このオーバーレイ画像には 256 種類の不透明カラーがあります。SOV 画像には 255 種類の不透明カラーと 1 種類の透明カラーがあります。このオプションは、ハードウェアによる透明カラーを有効にするため、SOV 画像を使用するウィンドウで、より高い性能が得られます。

-g *gamma-correction value*

このオプションは FFB2+ のみで使用することができます。このオプションによって、ガンマ補正の値を変更することができます。すべてのリニア画像ではガンマ補正を使用することができます。デフォルトでは、ガンマ補正の値は 2.22 です。0 より小さい値は無効 (不正) です。ガンマ補正の値はリニア画像に適用され、リニア画像の有効ガンマ値は 1.0 になります。これは、`XSolarisGetVisualGamma(3)` によって返される値です。この機能については、`XSolarisGetVisualGamma(3)` を参照してください。

このオプションは、ウィンドウシステムが稼動しているときに使用することができます。ガンマ補正の値を変更すると、リニア画像を使用して表示されているすべてのウィンドウが影響を受けます。

-gfile *gamma-correction file*

このオプションは FFB2+ のみで使用することができます。このオプションは、指定されたファイルからガンマ補正表を読み込みます。このファイルは、各行が R、G、B チャンネルの値を持つように書式化されている必要があります。それらの値は、16 進数で指定し、値と値の間は 1 つ以上の空白文字で区切ります。このファイルでは、3 つの値の組が 256 種類定義されます。このファイルの例を以下に示します。

```
0x00 0x00 0x00
0x01 0x01 0x01
0x02 0x02 0x02
...
...
```

```
0xff 0xff 0xff
```

このオプションを使用することによって、ウィンドウシステムが稼動しているときにガンマ補正表を読み込むことができます。新しいガンマ補正は、このリニア画像によって表示されているすべてのウィンドウに影響を与えます。ユーザーが指定した表によってガンマ補正を行う際は、ガンマ補正の値は定義されません。デフォルトでは、ウィンドウシステムはガンマ補正值として2.22を使用し、このガンマ補正值に対応してウィンドウシステムが作成したガンマ補正表を読み込みます。

-defaults

すべてのオプションの値をそれぞれのデフォルト値に戻します。

-propt

-file オプションで指定された OWconfig ファイルに書かれた FFB オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。ffbconfig の呼び出しが終了した後に、OWconfig ファイルに書き込まれるオプションの値を表示します。次に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/ffb0 ---
OWconfig: machine
Video Mode: NONE
Default Visual: Non-Linear Normal Visual
Visual Ordering: Linear Visuals are last
                  Overlay Visuals are last
OpenGL Visuals: disabled
SOV: disabled
Allocated WIDs: 32
```

-prconf

FFB のハードウェア構成を表示します。次に表示例を示します。

```
--- Hardware Configuration for /dev/fbs/ffb0 ---
Type: double-buffered FFB2 with Z-buffer
Board: rev x
PROM Information: @(#)ffb2.fth x.x xx/xx/xx
FBC: version x
DAC: Brooktree 9068, version x
3DRAM: Mitsubishi 1309, version x
EDID Data: Available - EDID version 1 revision x
Monitor Sense ID: 4 (Sun 37x29cm RGB color monitor)
Monitor possible resolutions: 1024x768x60, 1024x768x70,
                              1024x768x75, 1152x900x66, 1152x900x76,
                              1280x1024x67, 1280x1024x76, 960x680x112s,
                              640x480x60
Current resolution setting: 1280x1024x76
```

-help

ffbconfig コマンド行のオプションと機能の概要を一覧で表示します。

デフォルト設定

ffbconfig コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。

ffbconfig(1M)

ウィンドウシステムを実行する際に、ffbconfig による FFB オプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルト値を以下に示します。

オプション	デフォルト値
-dev	/dev/fbs/ffb0
-file	machine
-res	none
-deflinear	false
-defoverlay	false
-linearorder	last
-overlayorder	last
-expvis	enabled
-sov	enabled
-maxwids	32

-res オプションのデフォルト値 none とは、ウィンドウシステムが実行された場合に、画面解像度がそのデバイスに現在プログラムされている表示モードになることを意味しています。

注：これによって、PROM によってデバイスの解像度を指定しているユーザーとの共用性が保てます。(gx などの)一部のデバイスでは、PROM が表示モードを指定する唯一の手段です。これは、デフォルトの FFB 表示モードは、最終的に PROM によって決まることを意味しています。

使用例 例 1 モニターの種類の変更

モニターの種類を、垂直周波数 76 Hz で解像度 1280 × 1024 に変更する例を以下に示します。

```
example% /usr/sbin/ffbconfig -res 1280x1024x76
```

ファイル /dev/fbs/ffb0 デバイス特殊ファイル

属性 次の属性については、attributes(5) を参照してください。

属性タイプ	属性値
使用条件	SUNWffbcf

関連項目 mmap(2), attributes(5), fbio(7I), ffb(7D)

OpenWindows デスクトップ・リファレンスマニュアルのマニュアルページ

GFXconfig(1M)

名前	pgxconfig, GFXconfig – PGX32 (Raptor GFX) グラフィックスアクセラレータの設定
形式	<pre>/usr/sbin/pgxconfig [-dev <i>device-filename</i>] [-res <i>video-mode</i> [try noconfirm nocheck]] [-file machine system] [-depth 8 24] [-24only] [-defaults] /usr/sbin/pgxconfig [-propt] [-prconf] /usr/sbin/pgxconfig [-help] [-res ?] /usr/sbin/pgxconfig [-i]</pre>
機能説明	<p>pgxconfig ユーティリティーは PGX32 (Raptor GFX) グラフィックスアクセラレータの設定用コマンドで、これには X11 ウィンドウシステムの一部を PGX32 (Raptor GFX) 用にデフォルト設定する機能も含まれます。以前のバージョンでは、このユーティリティーは GFXconfig という名称でした。</p> <p>形式の項に記された pgxconfig の第 1 の形式では、指定したオプションを OWconfig ファイルに保存します。次回の PGX32 (Raptor GFX) デバイス上でのウィンドウシステム起動時に、ここで指定したオプションに従って PGX32 (Raptor GFX) デバイスを初期化します。OWconfig ファイルに保存されたオプションの更新内容は、異なる複数のウィンドウシステムセッションや再起動後のシステムでも有効となります。</p> <p>その他の形式 (2、3、4 番目の形式) を使用した場合はオプション -prconf, -propt, -help, -res ? を呼び出すだけで、OWconfig ファイルに保存されているオプションを更新することはありません。更に、3 番目の形式を使用した場合はこれ以外のオプションはすべて無視されます。</p> <p>-i オプションを使用すると、pgxconfig は、対話型モードで起動します。</p> <p>一度にオプションを指定できる PGX32 (Raptor GFX) デバイスの数は、1 つだけです。複数の PGX32 (Raptor GFX) デバイスにオプションを指定する場合は、pgxconfig -i を複数回起動する必要があります。</p> <p>pgxconfig コマンドで指定できるのは、PGX32 (Raptor GFX) 固有のオプションだけです。デフォルトの表示色数、デフォルトのビジュアルクラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください。SUNWxwman パッケージに含まれている Xsun(1) マニュアルページを参照してください。</p> <p>ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは /usr/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/etc/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。</p> <p>いずれの標準 OWconfig ファイルでも、書き込み権限があるのはスーパーユーザーだけです。したがって、pgxconfig は、スーパーユーザーがプログラムの所有者となり、setuid による root 権限で実行されます。</p>

オプション

`-dev device-filename`
 PGX32 (Raptor GFX) 特殊ファイルを指定します。デフォルトは `/dev/fbs/gfxp0`、または使用可能な場合であれば `/dev/fbs/raptor0` です。

`-file machine | system`
 更新する OWconfig ファイルを指定します。ここで `machine` が指定された場合は、`/etc/openwin` ディレクトリツリーにあるマシン固有の OWconfig ファイルが使用されます。`system` が指定された場合は、`/usr/openwin` ディレクトリツリーにある共通の OWconfig ファイルが使用されます。ファイルがない場合は、新たに生成されます。

`-res video-mode [try | noconfirm | nocheck]`
 PGX32 (Raptor GFX) デバイスに接続されているモニターの制御に使用する組み込みのビデオモードを指定します。

ビデオモードの形式は、以下のいずれか 1 つになります。

`widthxheightxrate` `width` はピクセル単位の画面の幅、`height` はピクセル単位の画面の高さ、`rate` は垂直方向の画面再描画周期です。`-res` では、再描画周期 `rate` の前の `x` は、`@` でも代用できます。つまり周期の指定に限っては、たとえば `1280x1024@76` のような形式にも対応しています。オプションに `-res ?` を付けて `pgxconfig` を実行すると (コマンド形式の項に記された 3 番目の形式)、ビデオモードの一覧が表示されます。ビデオボードとモニターの両方が、すべての解像度をサポートしているわけではありません。`noconfirm` または `nocheck` オプションを指定しないで、ボードのサポートしていない解像度を入力した場合には、`pgxconfig` はその要求を許可しません。また、`nocheck` オプションを指定しないで、モニターのサポートしていない解像度を指定した場合には、その値を適用する前に確認を求めてきます。

記号名 (Symbolic names) 便宜上、一部のビデオモードには記号名が定義されています。`widthxheightxrate` の形式の代わりに、記号名を `-res` の引数として指定することができます。記号名 `none` は、ウィンドウシステムを実行すると、画面の解像度が現在デバイスにプログラムされているビデオモードになることを意味します。

<code>svga</code>	<code>1024x768x60</code>
<code>1152</code>	<code>1152x900x76</code>
<code>1280</code>	<code>1280x1024x76</code>
<code>vga</code>	<code>640x480x60</code>
<code>none</code>	デフォルトのコンソールの解像度

GFXconfig(1M)

-res オプションには、ビデオモードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定できます。

noconfirm -res オプションを指定した際に、システムが使用可能であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあつた場合などに発生します。このような事態を避けるために `pgxconfig` のデフォルトの動作では、この問題についての警告メッセージと、処理を継続するかどうかを確認するメッセージを表示します。noconfirm オプションを指定すると、`pgxconfig` コマンドはこの確認をせずに、要求のあつたビデオモードにプログラムします。このオプションは、`pgxconfig` がシェルスクリプトから実行されている場合に便利です。

nocheck このオプションを指定すると、モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定されたビデオモードは、現在接続されているモニターに適切かどうかにかかわらず受け付けられます。このオプションは、PGX32 (Raptor GFX) デバイスに異なるモニターを接続する場合に便利です。このオプションの指定は、noconfirm の指定も兼ねます。

try このオプションを指定すると、指定したビデオモードを適用する前にテストすることができます。まず、指定したモードに基づいたテストパターンが表示されます。テストパターンが正常に表示された場合は、「y」(次いでキャリッジリターン)を入力します。「y」以外の文字を入力(次いでキャリッジリターンを入力)した場合は、「no」となります。

-res ?
PGX32 およびモニターがサポートする解像度の一覧を表示します。

-24only
Openwindows の実行時に、PGX32 (Raptor GFX) デバイスが 24 ビットカラーのみを使用するようにします。

-defaults
すべてのオプションの値をそれぞれのデフォルト値に戻します。

-propt
-file オプションで指定された `OWconfig` ファイルに書かれた PGX32 (Raptor GFX) オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。`pgxconfig` の呼び出しが終了した後に、`OWconfig` ファイルに書き込まれるオプションの値を表示します。以下に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/gfxp0 ---  
OWconfig: machine  
Video Mode: not set
```

-prconf

PGX32 (Raptor GFX) のハードウェア構成を表示します。以下に表示例を示します。

```

--- Hardware Configuration for /dev/fbs/gfxp0 ---
DAC: version 0x0
Type:
Board:
PROM: version 0x0
PROM Information:
RAM:
EDID Data:
Monitor Sense ID:
Card possible resolutions:  640x480x60, 800x600x75, 1024x768x60
                            1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76
                            1280x1024x60, 1152x900x66, 1152x900x76, 1280x1024x67
                            960x680x112S, 960x680x108S, 640x480x60i, 768x575x50i,
                            1280x800x76, 1440x900x76, 1600x1000x66, 1600x1000x76,
                            vga, svga, 1152, 1280, stereo, ntsc, pal
Monitor possible resolutions: 720x400x70, 720x400x88, 640x480x60
                              640x480x67, 640x480x72, 640x480x75, 800x600x56,
                              800x600x60, 800x600x72, 800x600x75, 832x624x75,
                              1024x768x87, 1024x768x60, 1024x768x70, 1024x768x75,
                              1280x1024x75, 1280x1024x76, 1152x900x66, 1152x900x76,
                              1280x1024x67, 960x680x112S, vga, svga, 1152, 1280
                              stereo
Current resolution setting: 1280x1024x76
Possible depths:
Current depth: 8

```

-help

pgxconfig コマンド行のオプションと機能の概要を一覧で表示します。

-i

pgxconfig コマンドを対話型モードで起動します。

デフォルト

pgxconfig コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。ただし、`-depth` と `-24only` については、その限りではありません。

ウィンドウシステムを実行する際に、pgxconfig による PGX32 (Raptor GFX) のオプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルト値は次のとおりです。

```

-dev          /dev/fbs/gfxp0
-file         system
-res         none

```

`-res` オプションのデフォルト値 `none` とは、ウィンドウシステムが実行された場合に、画面解像度とそのデバイスに現在プログラムされているビデオモードになることを意味しています。

GFXconfig(1M)

使用例	<p>例 1 モニターの種類を変更する。</p> <p>モニターの種類を、垂直周波数 76 Hz で解像度 1280 x 1024 に変更する例を以下に示します。</p> <pre>example% /usr/sbin/pgxconfig -res 1280x1024x76</pre>
ファイル	<pre>/dev/fbs/gfxp0</pre> <p>デバイス特殊ファイル</p> <pre>/usr/openwin/server/etc/OWconfig</pre> <p>システム構成ファイル</p> <pre>/etc/openwin/server/etc/OWconfig</pre> <p>マシン構成ファイル</p>
関連項目	<p>PGX32 PCI グラフィックスカード インストールマニュアル</p>

名前	installer – Solaris Web Start インストールユーティリティ	
形式	installer [-locales <i>list</i>] [-nodisplay] [-debug]	
機能説明	<p>installer ユーティリティは、Solaris Web Start のウィザードを起動します。ユーザーは、表示される一連のウィザードの指示に沿ってインストールを行います。</p> <p>installer ユーティリティは、Solaris に同梱されている別ソフトウェアの CD のトップディレクトリに含まれています。</p> <p>installer が含まれている CD にデスクトップのファイルマネージャからアクセスしている場合は、installer のアイコンをダブルクリックすることによって、ウィザードを起動することができます。スーパーユーザーになっていない場合は、スーパーユーザーのパスワードを入力するように指示メッセージが表示されます。</p> <p>installer ユーティリティは、他の UNIX スクリプトから実行することもできます。スクリプトから実行する場合は通常、installer ユーティリティを -nodisplay オプション付きで実行するようにします。</p>	
オプション	<p>次のオプションを指定できます。</p> <p>-locales <i>list</i> インストールする製品のロケールを選択します。<i>list</i> に指定されたロケールがインストールメディアに提供されていれば、そのロケールの製品がインストールされます。ロケールは、-locales オプションの後にカンマで区切って指定します。たとえば次のように指定すると、</p> <p style="padding-left: 40px;">installer -locales fr,de,it フランス語 (fr)、ドイツ語 (de)、イタリア語 (it) ロケールの製品がインストールされます。</p> <p>-nodisplay GUI (グラフィカルユーザーインターフェース) を使用せずにインストールを行います。-locales オプションでロケールを指定していない場合は、デフォルトのロケール (英語) の製品をインストールします。</p> <p>-debug インストール中の処理状況を示す情報を出力します。このオプションは、おもにインストール処理を診断したいときに使用します。</p>	
ファイル	/var/sadm/install/logs	インストールログファイルの保存ディレクトリ
関連項目	prodreg(1M)	

lpmove(1M)

名前	lpmove – 印刷要求の移動
形式	lpmove <i>request-ID destination</i> lpmove <i>destination1 destination2</i>
機能説明	<p>lpmove コマンドは、lp(1) または lpr(1B) によって待ち行列に入れられた印刷要求を宛先間で移動します。lpmove は、ローカルプリンタ上のジョブを移動する場合にだけ使用してください。</p> <p>第 1 の形式では、指定の印刷要求 (<i>request-ID</i>) を、指定の宛先 (<i>destination</i>) に移します。</p> <p>第 2 の形式では、ある宛先 (<i>destination1</i>) から他の宛先 (<i>destination2</i>) にすべての印刷要求を移します。この形式では lpmove は、新たに発生する <i>destination1</i> 宛の印刷要求も受け付けなくなります。</p> <p>要求を移動する際、lpmove は、印刷要求の移動先の受け入れ状態をチェックしません (accept(1M) を参照)。なお、オプション (たとえば、内容タイプ、特定の用紙の要求) 付きの要求に関しては、新たな宛先がそのオプションを扱えなければ、lpmove は移動を行いません。</p>
オペランド	<p>以下のオペランドを指定できます。</p> <p><i>destination</i> lpmove が指定された印刷要求を移動するプリンタ名またはプリンタのクラス名 (lpadmin(1M) を参照)。名前、POSIX スタイル名 (<i>server:destination</i>)、またはフェデレーテッド・ネーミング・サービス (FNS) 名 (.../service/printer/...) を使用して、<i>destination</i> を指定します。名前や FNS 名の命名規約については printers.conf(4) を参照してください。</p> <p><i>destination1</i> lpmove がすべての印刷要求を移動する移動元の名前。名前、POSIX スタイル名 (<i>server:destination</i>)、またはフェデレーテッド・ネーミング・サービス (FNS) 名 (.../service/printer/...) を使用して、<i>destination</i> を指定します。名前や FNS 名の命名規約については printers.conf(4) を、POSIX については standards(5) を参照してください。</p> <p><i>destination2</i> lpmove がすべての印刷要求を移動する移動先の名前。名前、POSIX スタイル名 (<i>server:destination</i>)、またはフェデレーテッド・ネーミング・サービス (FNS) 名 (.../service/printer/...) を使用して、<i>destination</i> を指定します。名前や FNS 名の命名規約については printers.conf(4) を参照してください。</p> <p><i>request-ID</i> 移動対象になる特定の印刷要求。lpstat が示す印刷要求に対応する識別子を <i>request-ID</i> に指定します。lpstat(1) のマニュアルページを参照してください。</p>
終了ステータス	<p>以下の終了ステータスが返されます。</p> <p>0 正常終了</p>

0 以外 エラーが発生した
 ファイル /var/spool/printd/印刷待ち行列
 属性 次の属性については `attributes(5)` のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWpcu

関連項目 `lp(1)`, `lpr(1B)`, `lpstat(1)`, `accept(1M)`, `lpadmin(1M)`, `lpsched(1M)`,
`printers.conf(4)`, `attributes(5)`, `standards(5)`
Solaris のシステム管理 (第 1 巻)

lpsched(1M)

名前	lpsched – LP 印刷サービスの起動				
形式	lpsched [-f <i>num_filters</i>] [-n <i>num_notifiers</i>] [-p <i>fd_limit</i>] [-r <i>reserved_fds</i>]				
機能説明	<p>lpsched コマンドは LP 印刷サービスを起動または再起動します。</p> <p>lpshut コマンドは LP 印刷サービスを停止します。lpsched でプリンタを再起動すれば、lpshut で印刷を中止した要求に関しては (始めから) 再印刷できます (lpshut(1M) を参照)。</p>				
オプション	<p>以下のオプションを指定できます。</p> <p>-f <i>num_filters</i> 印刷サーバー上で実行できる、並行するスローフィルタ数を指定します。何も指定しない場合は、デフォルトとして 1 が使用されます。サーバーの構成によっては値を 1 にすると、サーバーにジョブの待ち行列が存在していても、プリンタがアイドル状態のままになります。</p> <p>-n <i>num_notifiers</i> 印刷サーバー上で実行できる、並行する通知プロセス数を指定します。何も指定しない場合は、デフォルトとして 1 が使用されます。</p> <p>-p <i>fd_limit</i> lpsched プロセスで使用するファイル記述子のリソース制限値を指定します。何も指定しない場合は、デフォルトとして 4096 が使用されます。非常に大型でアクティブな印刷サーバー上では、この値を増やす必要はありません。</p> <p>-r <i>reserved_fds</i> 大量ロード下で、スケジューラが内部通信に予約するファイル記述子の数を指定します。何も指定しない場合は、デフォルトとして 2 が使用されます。高速ロード下での問題を障害追跡する場合に、その解析を指示しない限りは、この値を修正する必要はありません。</p>				
終了ステータス	<p>以下の終了ステータスが返されます。</p> <p>0 正常終了</p> <p>0 以外 エラーが発生した</p>				
ファイル	/var/spool/lp/ *LP 印刷待ち行列				
属性	次の属性については attributes(5) のマニュアルページを参照してください。				
	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">属性タイプ</th> <th style="text-align: center;">属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWpsu</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWpsu
属性タイプ	属性値				
使用条件	SUNWpsu				
関連項目	lp(1), lpstat(1), lpadmin(1M), lpmove(1M), lpshut(1M), attributes(5)				

lpshut(1M)

名前 lpshut - LP 印刷サービスの停止

形式 **lpshut**

機能説明 lpshut コマンドは LP 印刷サービスを停止します。

lpshut が実行されると、プリンタはただちに印刷を中止します。プリンタを起動または再起動するには、lpsched(1M) を使用します。

終了ステータス 以下の終了ステータスが返されます。

0 正常終了

0 以外 エラーが発生した

ファイル /var/spool/lp/印刷待ち行列

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWpsu

関連項目 lp(1), lpstat(1), lpadmin(1M), lpmove(1M), lpsched(1M), attributes(5)

『Solaris のシステム管理 (第 1 巻)』

名前	luxadm – Sun StorEdge A5000、RSM、SPARCstorage Array (SSA)サブシステム、Sun Fire 880 内部記憶装置サブシステムおよび各 Fiber Channel Arbitrated Loop (FC_AL) デバイスの管理プログラム
形式	luxadm [<i>options...</i>] <i>subcommand</i> [<i>options...</i>] <i>enclosure</i> [, <i>dev</i>] <i>pathname...</i>
機能説明	<p>luxadm プログラムは、Sun StorEdge A5000、RSM、SPARCstorage Array サブシステム、Sun Fire 880 内部記憶装置サブシステムおよび各 FC_AL デバイスの管理コマンドです。luxadm は、コマンド行に指定される引数やオプションに応じて、さまざまな制御処理や照会処理を実行します。</p> <p>コマンド行にはサブコマンドの指定が必要です。コマンド行には、サブコマンドに応じてオプションやパラメタを指定することができます。オプションには、通常 1 つ以上の格納装置名またはパス名を指定します。指定する文字数は、サブコマンドを一意に識別するのに必要なだけでかまいません。</p> <p>サブコマンドが作用するデバイスをパス名で指定します。Sun StorEdge A5000 サブシステムでは、パス名の代わりに、デバイスまたはデバイスに対するポートのワールドワイド名 (WWN) を入力することによってディスクデバイスまたは格納装置サービスコントローラを指定することができます。また、Sun StorEdge A5000 の格納装置名および格納装置内の特定のデバイスを示す識別子 (オプション) を入力することによってデバイスを指定することもできます。各 FC_AL デバイスを指定するには、デバイスの WWN またはポートを入力します。</p>
パス名	<p>デバイスまたはコントローラを、完全な物理パス名または完全な論理パス名で指定します。</p> <p>Sun StorEdge A5000 に対するデバイスの一般的な物理パス名を以下に示します。</p> <pre>/devices/sbus@1f,0/SUNW,socal@1,0/sf@0,0/ssd@w2200002037000f96, 0:a,raw</pre> <p>または</p> <pre>/devices/io-unit@f,e0200000/sbi@0,0/SUNW,socal@2,0/sf@0,0/ssd@34, 0:a,raw</pre> <p>システム上のすべての Sun StorEdge A5000 IB (Interface Board) および、Sun Fire 880 SES デバイスコントローラに対しては、物理パスへの論理リンクが /dev/es ディレクトリに格納されます。論理リンクの例として、/dev/es/ses0 などがあります。</p> <p>FC_AL デバイスや Sun StorEdge A5000 サブシステム IB または、Sun Fire 880 内部記憶装置サブシステムを選択するために、パス名の代わりに WWN を使用することもできます。WWN は、デバイスを使用するためのポートまたはデバイスそのものを表す 16 桁の 16 進数の値です。一般的な WWN の値を次に示します。</p> <pre>2200002037000f96</pre> <p>WWN の形式については、「注意事項」を参照してください。</p>

luxadm(1M)

SPARCstorage Array コントローラに対する一般的な物理パス名を次に示します。

```
/devices/. . . /. . . /SUNW,soc@3,0/SUNW,pln@  
xxxxxxx,xxxxxxx:ctlr
```

RSM の一般的な物理パス名を以下に示します。

```
/devices/sbus@1f,0/QLGC,isp@1,10000:devctl
```

SPARCstorage Array または RSM コントローラの指定を簡単にするために、cN の形式による論理パス名が使用可能になっています。Nには、論理コントローラ番号を指定します。luxadm は、SPARCstorage Array または RSM コントローラに接続されているディスクの /dev/rdisk ディレクトリの中にあるエントリを見つけるために、cN 名を使用します。次に、/dev/rdisk エントリは SPARCstorage Array または RSM コントローラの物理名を調べるために使用されます。

SPARCstorage Array ディスクに対する一般的なパス名を次に示します。

```
/devices/. . . /. . . /SUNW,soc@3,0/SUNW,  
pln@xxxxxxx,xxxxxxx/ssd@0,0:c,raw
```

次に、一般的な論理パス名を示します。

```
/dev/rdisk/c1t0d0s2
```

RSM の一般的な物理パス名を以下に示します。

```
/devices/sbus@1f,0/QLGC,isp@1,10000/sd@8,0:c,raw
```

次に、一般的な論理パス名を示します。

```
/dev/rdisk/c2t8d0s2
```

Sun Fire 880 内部記憶装置サブシステムのディスクの一般的な物理パス名を示します。

```
/devices/pci@8,600000/SUNW,qlc@2/fp@0,0/ssd@w2100002037a6303c,0:a
```

次に、一般的な論理パス名を示します。

```
/dev/rdisk/c2t8d0s2
```

各 FC_AL デバイスの一般的なパス名を示します。

```
/devices/sbus@3.0/SUNW,socal@d,10000/sf@0,0/ssd@w2200002037049fc3,0:a,raw次に、  
一般的な論理パス名を示します。
```

```
/dev/rdisk/c1t0d0s2
```

格納装置 Sun StorEdge A5000 では、デバイスは格納装置名とスロット名で特定することができます。

```
box_name[, fslot_number]  
box_name[, rslot_number]
```

box_name は、Sun StorEdge A5000 の格納装置名で、*enclosure_name* サブコマンドで指定します。オプションの *slot_number* パラメタを指定しないと、*box_name* には、Sun StorEdge A5000 サブシステム IB が指定されます。

f または *r* は、Sun StorEdge A5000 の格納装置の前面のスロット(*f*)か、背面のスロット(*r*)かを指定します。

slot_number は、Sun StorEdge A5000 の格納装置のデバイスのスロット番号を指定します。指定することができる番号の範囲は 0～6 または 0～10 です。

Sun Fire 880 内部記憶装置サブシステムでも、デバイスは格納装置名とスロット名で特定することができますが、ディスクは、1 セットしかありません。

```
box_name[, sslot_number]
```

box_name は、Sun Fire 880 の格納装置名で、*enclosure_name* サブコマンドで指定します。オプションの *slot_number* パラメタを指定しないと、*box_name* には、Sun Fire 880 内部記憶装置サブシステムの格納装置サービスデバイスが指定されます。*slot_number* は、Sun Fire 880 内部記憶装置サブシステムのディスクスロット番号を指定します。指定することができる番号の範囲は 0～11 です。

ディスクおよびサブシステムの論理名については、[disks\(1M\)](#) および [devlinks\(1M\)](#) を参照してください。

オプション 次のオプションはすべてのサブコマンドに対して指定することができます。

```
-e      エキスパート (Expert) モードです。このオプションは初心者の方にはお勧め  
        しません。  
-v      詳細表示 (Verbose) モードです。
```

特定のサブコマンドだけに指定するオプションについては、「使用方法」の各サブコマンドの説明を参照してください。

オペランド 次のオペランドを指定することができます。

luxadm(1M)

	<i>enclosure</i>	Sun StorEdge A5000 または Sun Fire 880 内部記憶装置サブシステム
	<i>pathname</i>	Sun StorEdge A5000 IB、Sun Fire 880 内部記憶装置サブシステム、SPARCstorage Array、または RSM コントローラの名前 (cN)、またはディスクデバイスの論理パスまたは物理パスを指定します。 <i>pathname</i> は、Sun StorEdge A5000 IB、Sun StorEdge A5000 ディスク、または各 FC_AL デバイスの WWN でも指定することができます。
使用方法		
サブコマンド	<code>display enclosure[<i>dev</i>]... <i>pathname</i>...</code> <code>display -p <i>pathname</i>...</code> <code>display -r enclosure[<i>dev</i>]... <i>pathname</i>...</code> <code>display -v enclosure[<i>dev</i>]... <i>pathname</i>...</code>	格納装置またはデバイス固有のデータを表示します。
		サブシステムのデータは、格納装置の環境検知情報、およびディスクの状態を含む、すべてのサブシステムのデバイスの状態で構成されます。
		ディスクデータは、照会、容量、および設定情報で構成されます。
	-p	<i>pathname</i> で指定されたデバイスまたはサブシステムの性能情報を表示します。このオプションは、性能情報を保存するサブシステムに対してのみ指定することができます。
	-r	<i>pathname</i> で指定された FC_AL デバイスのエラー情報を表示します。また、パス名が Sun StorEdge A5000 の場合は、そのループ上のすべてのデバイスのエラー情報を表示します。-r オプションは Sun StorEdge A5000 サブシステムまたは各 FC_AL デバイスに対してのみ指定することができます。
	-v	モード検知データを含む、詳細表示モードで表示します。
	<code>download [-s] [-w WWN] [-f <i>filename_path</i>] enclosure...</code>	<i>prom</i> イメージを、 <i>filename_path</i> で指定された Sun StorEdge A5000 サブシステムのインタフェースボード装置か、 <i>enclosure</i> または <i>pathname</i> で指定された Sun Fire 880 内部記憶装置サブシステムまたは SPARCstorageArray コントローラにダウンロードします。SPARCstorage Array は、ダウンロードコードを使用するためにリセットする必要があります。
		Sun StorEdge A5000 のダウンロードが終了すると、Sun StorEdge A5000 はリセットされ、ダウンロードコードが実行されます。ファイル名が指定されていない場合は、デフォルトの <i>prom</i> イメージが使用されます。SPARCstorage Array コントローラ用のデフォルトの <i>prom</i> イメージは、

/usr/lib/firmware/ssa/ssafirmware ディレクトリに格納されます。Sun StorEdge A5000 用のデフォルトの prom イメージは /usr/lib/locale/C/LC_MESSAGES ディレクトリに ibfirmware というファイル名で格納されます。

Sun Fire 880 内部記憶装置サブシステムのダウンロードが終了すると、サブシステムはリセットされ、ダウンロードコードが実行されます。Sun Fire 880 内部記憶装置サブシステム用のデフォルトのファームウェアイメージは /usr/platform/SUNW,Sun-Fire-880/lib/images/int_fcbpl_fw ディレクトリに格納されます。

-s

保存オプションです。-s オプションはダウンロードしたファームウェアを FEPRROM に保存します。-s を省略すると、ダウンロードしたファームウェアは保存されず、電源の再投入後に消えてしまいます。

-s オプションは SPARCstorage Array コントローラには適用されません。SPARCstorage Array コントローラは、常に FEPRROM にファームウェアをダウンロードするためです。

-s オプションは Sun Fire 880 内部記憶装置サブシステムには適用されません。Sun Fire 880 内部記憶装置サブシステムは、常にフラッシュメモリにファームウェアをダウンロードするためです。

-s オプションを使用すると、ダウンロードサブコマンドがサブシステム上の FEPRROM を変更するため、-s オプションの使用には注意が必要です。

-w WWN

SPARCstorage Array コントローラのワールドワイド名を変更します。WWN は、先頭が 0 で始まる 12 桁の 16 進数です。-w オプションは、SPARCstorage Array に対してのみ指定することができます。新しい SPARCstorage Array コントローラのイメージは、8 バイトのワールドワイド名のうち、下位の 6 バイトが WWN で指定された値に変更されたものになります。

enclosure_name new_name enclosure | pathname

enclosure または pathname で指定された単数または複数の格納装置名を変更します。新しい名前は 16 文字以下の英数字で new_name に指定します。このサブコマンドは Sun StorEdge A5000 および Sun Fire 880 内部記憶装置サブシステムに対してだけ指定することができます。

fc_s_download [-F] [-f fcode-file]

fcode-file で指定されたファイルに含まれる fcode をすべての FC/S SBus カードにダウンロードします。このコマンドは対話型で、fcode をダウンロードする前にユーザーに対して確認を促します。

fc_s_download は、シングルユーザーモードでだけ使用してください。入出力操作が行われているホストアダプタに対して、そのアダプタを更新する目的で fc_s_download を指定すると、アダプタのリセットの原因となります。新しく更新した FCode は、システムを再起動したときに実行され、表示できるようになります。

luxadm(1M)

-f *fcode-file*

-f *fcode-file* オプションを省略すると、各 FC/S SBus カードの中の現在のバージョンの *fcode* が表示されます。

-F

オプションを指定すると、*fcode* は強制的にダウンロードされますが、ダウンロードの前にユーザーに対して確認を促します。現在のバージョンのオペレーションシステムと共にリリースされた FC/S SBus カード *fcode* のバージョンは `/usr/lib/firmware/fc_s` ディレクトリに `fc_s_fcode` の名前で格納されています。

`fcals_download [-f fcode-file]`

fcode-file に指定されたファイルに含まれる *fcode* をすべての FC100/S SBus カードにダウンロードします。このコマンドは対話型で、*fcode* をダウンロードする前にユーザーに対して確認を促します。

`fcals_download` は、シングルユーザーモードでだけ使用してください。入出力操作があるホストアダプタに対して、そのアダプタを更新する目的で `fcals_download` を指定すると、アダプタのリセットの原因となります。新しく更新した *fcode* は、システムを再起動したときに実行され、表示できるようになります。

-f *fcode-file*

-f オプションを省略すると、各 FC100/S SBus カード中の現在のバージョンの *fcode* が表示されます。

`fcode_download -p`

`fcode_download -d dir-name`

インストール済みの FC/S、FC100/S、FC100/P、または FC100/2P ホストバスアダプタカードを検出し、*dir-name* の *fcode* を適切なカードにダウンロードします。このコマンドは対話型で、各ファイルタイプに対して適切なカードを決定します。各デバイスに *fcode* をダウンロードする前にユーザーに対して確認を促します。

`fcode_download` は、シングルユーザーモードで *fcode* だけを読み込むときに使用します。入出力操作があるホストアダプタに対して、そのアダプタを更新する目的で `fcode_download` を指定すると、アダプタのリセットの原因となります。新しく更新した *fcode* は、システムを再起動したときに実行され、表示できるようになります。

-d *dir-name*

dir-name ディレクトリに格納されている *fcode* ファイルを適切なアダプタカードにダウンロードします。

-p

各カードに読み込まれている現在のバージョンの *fcode* を表示します。ダウンロードは行われません。

`inquiry enclosure[dev]... | pathname...`

`enclosure` または `pathname` で指定されたデバイスに対する照会情報を表示します。

`insert_device [enclosure,dev ...]`

このコマンドを使用して、電源を入れたまま 1 つまたは複数のデバイスを追加することができます。ホットプラグ操作中の制限に関しては「注意事項」を参照してください。このサブコマンドは Sun StorEdge A5000、Sun Fire 880 内部記憶装置サブシステム、RSM および各 FC_AL デバイスに対してのみ指定することができます。Sun StorEdge A5000 に対して複数の格納装置が指定された場合は、複数のバスに対して同時に追加が行われます。このサブコマンドの引数を省略すると、すべての格納装置または各 FC_AL デバイスが追加されます。RSM に対して指定することができます。Sun StorEdge A5000 または Sun Fire 880 内部記憶装置サブシステムに対しては、このサブコマンドは、電源を入れたまま 1 つまたは複数のデバイスを追加するすべての過程をユーザーとの対話によって行います。複数のディスクが指定された場合は、それらのディスクが正しいかどうかの確認が行われ、ユーザーは継続するか中止するかを選択することができます。その後で、ディスクまたは格納装置の追加を実行するかどうかの確認が行われ、それらのデバイスの論理パス名が作成および表示されます。

RSM に対しては、以下の手続きが行われます。

- バスを休止させます。
- デバイスが安全に追加できることをユーザーに通知します。
- デバイスが追加されたことを確認するメッセージを表示します。
- バスを休止解除します。
- 新規デバイスに対する論理デバイス名を作成します。

`led enclosure,dev ... | pathname ...`

`enclosure` または `pathname` で指定されたディスクに対応している LED の現在の状態を表示します。このサブコマンドはこの機能を持つサブシステムに対してのみ指定することができます。

`led_blink enclosure,dev ... | pathname ...`

`enclosure` または `pathname` で指定されたディスクに対応している LED の点滅を開始するようサブシステムに指示します。このサブコマンドはこの機能を持つサブシステムに対してのみ指定することができます。

`led_off enclosure,dev ... | pathname ...`

`enclosure` または `pathname` で指定されたディスクに対応している LED を消灯させるようサブシステムに指示します。Sun StorEdge A5000 サブシステムでは、LED の消灯や点滅の停止ができる状態とできない状態があります。『Sun StorEdge A5000 設置・サービスマニュアル』(805-4111)を参照してください。このサブコマンドはこの機能を持つサブシステムに対してのみ指定することができます。

`led_on pathname ...`

`pathname` で指定されたディスクに対応している LED を点灯させるようサブシステムに指示します。このサブコマンドはこの機能を持つサブシステムに対してだけ指定することができます。

`power_off [-F] enclosure[,dev] ... | pathname ...`

`power_off pathname [enclosure-port] ... | controller tray-number`

Sun StorEdge A5000 に対してこのオプションを指定した場合は、Sun StorEdge A5000 サブシステムが省電力モードに切り替わります。Sun StorEdge A5000 ドライブは、省電力モードでは使用することができません。SPARCstorage Array 中の

格納装置サービスカードに対してこのオプションを指定した場合は、RSM トレーの電源が切れます。Sun StorEdge A5000 中のドライブに対してこのオプションを指定した場合は、そのドライブがドライブ off/unmated モードに設定されます。ドライブ off/unmated 状態では、ドライブが停止し、バイパスモードになります。このコマンドは、Sun Fire 880 内部記憶装置サブシステムには適用されません。

-F

強制オプションは Sun StorEdge A5000 のみに適用されます。このオプションを指定すると、luxadm は、1 つまたは複数のデバイスを、それらがホストによって使用されていても電源切断しようとしています。

警告：現在使用されているデータを含むデバイスの電源を切断すると、予想可能な結果を引き起こします。デバイスの電源を切断する際は、まず、通常の方法を (-F を指定せずに) 試してください。このオプションは、通常の確認を無効にすることによる結果を理解した上で使用してください。

power_on enclosure [, dev] ..

Sun StorEdge A5000 に対してこのオプションを指定した場合は、Sun StorEdge A5000 サブシステムが省電力モードから抜けません。Sun StorEdge A5000 ドライブは、省電力モードでは使用することができません。SPARCstorage Array RSM トレーの電源をプログラムによって投入する方法はありません。Sun StorEdge A5000 中のドライブに対してこのオプションを指定した場合は、そのドライブが通常の起動状態に設定されます。このコマンドは、Sun Fire 880 内部記憶装置サブシステムには適用されません。

probe [-p]

接続された Sun StorEdge A5000 サブシステム、Sun Fire 880 内部記憶装置サブシステムおよび各 FC_AL デバイスすべてに関する情報を検索して表示します。この情報には、論理パス名、WWN このサブコマンドは、同じ格納装置名を持つ異なる Sun StorEdge A5000 を見つけると、ユーザーに対して警告を發します。

-p

物理パス名を表示に含めます。

qlgc_s_download [-f fcode-file]

fcode-file ファイルに格納されている fcode をすべての FC100/P、FC100/2P PCI ホストアダプタカードにダウンロードします。このコマンドは対話型で、fcode を各ドライブにダウンロードする前にユーザーに対して確認を促します。

qlgc_s_download は、シングルユーザーモードでだけ使用してください。入出力操作があるホストアダプタに対して、そのアダプタを更新する目的で fcal_s_download を指定すると、アダプタのリセットの原因となります。新しく更新した fcode は、システムを再起動したときに実行され、表示できるようになります。

-f fcode-file

-f オプションを省略すると、各 FC100/P、FC100/2P PCI カードの、現在のバージョンの fcode が表示されます。

release *pathname*

指定されたディスクの予約を解除します。パス名は、ディスクの物理または論理パス名でなければなりません。SPARCstorage Array コントローラのパス名を指定した場合、SPARCstorage Array 中のすべてのディスクの予約が解除されます。

このサブコマンドは、履歴および診断目的以外では使用しないでください。

remove_device [-F] *enclosure[,dev]... | pathname ...*

このコマンドを使用して、電源を入れたまま 1 つまたは複数のデバイスを削除することができます。このサブコマンドはすべての格納装置を削除する場合にも指定することができます。このサブコマンドは Sun StorEdge A5000、Sun Fire 880 内部記憶装置サブシステム、RSM、および各 FC_AL デバイスに対して指定することができます。ホットプラグ操作中の制限に関しては「注意事項」を参照してください。Sun StorEdge A5000、Sun Fire 880 内部記憶装置サブシステム および各 FC_AL デバイスに対しては、このサブコマンドは、電源を入れたまま 1 つまたは複数のデバイスを削除するすべての過程をユーザーとの対話によって行います。複数のディスクが指定された場合は、それらのディスクが正しいかどうかの確認が行われ、ユーザーは継続するか中止するかを選択することができます。その後で、ディスクまたは格納装置の削除を実行するかどうかの確認が行われ、それらのデバイスの論理パス名が作成および表示されます。

RSM に対しては、以下の手続きが行われます。

- デバイスをオフラインにします。
- バスを休止させます。
- デバイスが安全に削除できることをユーザーに通知します。
- デバイスが削除されたことを確認するメッセージを表示します。
- バスを休止解除します。
- (取り外された) デバイスをオンラインに戻します。
- 削除されたデバイスの論理デバイス名を削除します。

多重ホストディスクに対しては、以下の手続きが行われます。

- `luxadm remove_device` コマンドを最初のホスト上で実行します。継続のプロンプトが表示されたら待機します。
- `luxadm remove_device` コマンドを他のホスト上で実行します。継続のプロンプトが表示されたら待機します。
- `remove_device` コマンドの実行を最初のホスト上で継続します。プロンプトが表示されたらこのデバイスを削除します。
- 他のホスト上での `luxadm remove_device` コマンドの実行を完了します。

-F

1 つまたは複数のデバイスに対してホットプラグを適用するように `luxadm` に指示します。これらのデバイスがホストで使用されている場合 (*busy* または *reserved* の場合) にも適用されます。ホットプラグ オペレーションが強制的に実行されます。

警告: 現在使用されているデータが格納されているデバイスを取り外すと、予期しない結果が発生します。通常は、`-F` を指定しないでホットプラグを適用します。`-F` は、通常のホットプラグ検査を無効にしたときの結果がわかっているときにだけ使用します。

replace_device [-F] *pathname*

このコマンドは RSM に対してのみ指定することができます。ホットプラグ操作中の制限に関しては「注意事項」を参照してください。このサブコマンドは、電源を入れたままデバイスを交換するすべての過程をユーザーとの対話によって行います。

RSM に対しては、以下の手続きが行われます。

- デバイスをオフラインにします。
- バスを休止させます。
- デバイスが安全に削除できることをユーザーに通知します。
- デバイスが削除されたことを確認するメッセージを表示します。
- バスを休止解除します。
- デバイスをオンラインに戻します。

-F

1 つまたは複数のデバイスに対してホットプラグを適用するように luxadm に指示します。これらのデバイスがホストで使用されている場合 (*busy* または *reserved* の場合) にも適用されます。ホットプラグ オペレーションが強制的に実行されます。

警告: 現在使用されているデータが格納されているデバイスを取り外すと、予期しない結果が発生します。通常は、**-F** を指定しないでホットプラグを適用します。**-F** は、通常のホットプラグ検査を無効にしたときの結果がわかっているときにだけ使用します。

reserve *pathname*

指定されたディスクを、ホストによる排他的利用のために予約します。パス名は、ディスクの物理または論理パス名でなければなりません。*pathname* が SPARCstorage Array コントローラのパス名の場合、SPARCstorage Array 中のすべてのディスクが予約されます。

このサブコマンドは、履歴および診断目的以外では使用しないでください。

set_boot_dev [-y] *pathname*

システム PROM 中のブートデバイス変数を、*pathname* で指定された物理デバイス名に設定します。指定する物理デバイス名は、ブロック型特殊デバイス、または起動ファイルシステムのマウント先のディレクトリのパス名です。通常、このコマンドは、PROM 中のデフォルト起動デバイスを設定するためにユーザーに対して確認を促します。**-y** オプションを指定すると、ユーザーに対して確認を要求しません。

start [-t *tray-number*] *pathname* ...

指定されたディスクを起動します。*pathname* に SPARCstorage Array コントローラを指定した場合は、SPARCstorage Array 中のすべてのディスクを起動します。

-t *tray-number* で指定された番号のトレイ中のすべてのディスクを起動します。*pathname* には SPARCstorage Array コントローラを指定する必要があります。

SPARCstorage
Array のサブコマ
ンド

```
stop [ -t tray-number ] pathname ...
  指定されたディスクを停止します。 pathname に SPARCstorage Array コントローラ
  を指定した場合は、SPARCstorage Array 中のすべてのディスクを停止します。

  -t
    tray-number で指定された番号のトレイ中のすべてのディスクを停止します。
    pathname には SPARCstorage Array コントローラを指定する必要があります。

fast_write [ -s ] -c pathname
fast_write [ -s ] -d pathname
fast_write [ -s ] -e pathname
  SPARCstorage Array の書き込みの性能を向上させるために NVRAM の利用を有
  効・無効にします。 pathname には SPARCstorage Array コントローラか、個々の
  ディスクを指定します。

  -s
    電源を切断する前に SPARCstorage Array が変更を保存するように指定します。

  -c
    同期書き込みの場合にのみ高速書き込みを有効にします。

  -d
    高速書き込みを無効にします。

  -e
    高速書き込みを有効にします。

nvram_data pathname
  指定されたディスクの NVRAM 中にある高速書き込みデータの量を表示しま
  す。このコマンドは 1 つのディスクに対してのみ使用することができます。

perf_statistics -d pathname
perf_statistics -e pathname
  指定された SPARCstorage Array コントローラに関する性能統計情報を保存するか
  どうかを指定します。性能統計情報の保存は、display -p サブコマンドを使用する
  前に有効になっている必要があります。このサブコマンドは、SPARCstorage
  Array コントローラに対してのみ指定することができます。

  -d
    性能統計情報の保存を無効にします。

  -e
    性能統計情報の保存を有効にします。

purge pathname
  あるディスクの NVRAM から高速書き込みデータをディスクに書き出します。コ
  ントローラを指定した場合は、そのコントローラ中のすべてのディスクが対象とな
  ります。このオプションの使用時には注意が必要で、通常はドライブに障害が発生
  した場合にだけ使用します。

sync_cache pathname
  指定されたディスクの NVRAM から、未処理の書き込みすべてをディスクに書き
  出します。 pathname にコントローラを指定した場合は、その SPARCstorage Array
  サブシステム中のすべてのディスクが操作の対象となります。
```

luxadm(1M)

格納装置サービス
カードのサブコマ
ンド

`env_display` および `alarm*` サブコマンドは、SPARCstorage Arrayの RSM トレー
の中にある格納装置サービスカード (SES) のみが対象となります。RSM トレーは、
SES デバイスの論理パス名または物理パス名で指定するか、コントローラにトレー番
号を続けて指定します。コントローラは `cN` または `SSA` コントローラの物理パス名で
指定します。

`alarm pathname | controller tray_number`
警告音の現在の状態を表示します。

`alarm_off pathname | controller tray_number`
指定された RSM トレーの警告音を無効にします。

`alarm_on pathname | controller tray_number`
指定された RSM トレーの警告音を有効にします。

`alarm_set controller-pathname | controller tray_number [seconds]`
警告音の長さを秒単位で指定します。

`env_display pathname | controller tray_number`
指定された装置の環境情報を表示します。

**Sun StorEdge
A5000、Sun Fire
880** 内部記憶装置サ
ブシステムおよび
各 **FC_AL** デバイス
のエキスパート
モードコマンド

以下のサブコマンドは、経験が豊富なユーザーのみが使用することができます。ま
た、Sun StorEdge A5000、Sun Fire 880 内部記憶装置サブシステムおよびファイバ
チャネルループのみを対象にしています。これらのコマンドは、Sun StorEdge A5000
サブシステムとファイバチャネルループの知識が豊富なユーザーだけが使用すること
ができます。

バスを操作するエキスパートサブコマンドにディスクを指定した場合は、そのサブコ
マンドは、指定したディスクに接続されたバスを操作します。

`-e forcelp enclosure[, dev] ... | pathname ...`
ループ初期化基本式 (LIP) の処理を使ってリンクを強制的に最初期化します。
`enclosure` または `pathname` には、ループ上のどのデバイスでも指定することがで
きます。複数のループ構成に対して特定のパスを指定する場合は、パス名を使用し
ます。

このコマンドは経験者向けのコマンドで、使用には注意が必要です。このコマンド
は、ループ上のすべてのポートをリセットします。

`-e rdls enclosure[, dev] ... | pathname ...`
`enclosure` または `pathname` で指定されたデバイスを含むループにある、すべての
使用可能なデバイスのリンクエラー状態情報を読み込み、表示します。

その他のエキス
パートモードサブ
コマンド

ホットプラグ操作中の制限に関しては「注意事項」を参照してください。以下のサブ
コマンドは、経験が豊富なユーザーのみが使用することができます。

これらのコマンドは、Sun Fire 880 内部記憶装置サブシステムには適用されません。

`-e bus_getstate pathname`
指定されたバスの状態を取得、表示します。

`-e bus_quiesce pathname`
指定されたバスを休止します。

```

-e bus_reset pathname
    指定されたバスのみをリセットします。
-e bus_resetall pathname
    指定されたバスおよびすべてのデバイスをリセットします。
-e bus_unquiesce pathname
    指定されたバスを休止解除します。
-e dev_getstate pathname
    指定されたデバイスの状態を取得、表示します。
-e dev_reset pathname
    指定されたデバイスをリセットします。
-e offline pathname
    指定されたデバイスをオフラインにします。
-e online pathname
    指定されたデバイスをオンラインにします。

```

使用例

例 1 システム上にあるすべての Sun StorEdge A5000 および FC_AL の表示

システム上にある Sun StorEdge A5000 および FC_AL デバイスを検索、表示する例を次に示します。

```
example% luxadm probe
```

例 2 SSA の表示

SSA を表示する例を次に示します。

```
example% luxadm display c1
```

例 3 Sun StorEdge A5000 または Sun Fire 880 内部記憶装置サブシステムの表示

Sun StorEdge A5000 または Sun Fire 880 内部記憶装置サブシステムを表示する例を次に示します。

```
example% luxadm display /dev/es/ses0
```

例 4 2 つのサブシステムの表示

格納装置名を使って 2 つのサブシステムを表示する例を次に示します。

```
example% luxadm display BOB system1
```

例 5 最初のディスクに関する情報の表示

BOB.f1 という名前の格納装置の前面にある最初のディスクに関する情報を表示する例を示します。前面のディスクを指定する場合には *f* を指定します。背面のディスクを指定する場合は *r* を指定します。

luxadm(1M)

例 5 最初のディスクに関する情報の表示 (続き)

```
example% luxadm display BOB,f0
```

例 6 Sun Fire 880 内部記憶装置サブシステムに関する情報の表示

Sun Fire 880 内部記憶装置サブシステムにはディスクセットが1つしかありません。この場合には、`-s` オプションを使用してスロットを指定してください。

```
example% luxadm display BOB,s0
```

例 7 Sun StorEdge A5000 ディスク、格納装置、または各 FC_AL デバイスに関する情報の表示

ポートの WWN が 2200002037001246 の Sun StorEdge A5000 ディスク、格納装置、または各 FC_AL デバイスに関する情報を表示する例を次に示します。

```
example% luxadm display 2200002037001246
```

例 8 サブコマンドとして一意に認識できるだけの文字までを入力する例

サブコマンドとして一意に認識できるだけの長さの文字列を使用する例を次に示します。

```
example% luxadm disp BOB
```

例 9 エラー情報の表示

格納装置 BOB があるループに関するエラー情報を表示する例を次に示します。

```
example% luxadm display -r BOB
```

例 10 インタフェースボードへの新しいファームウェアのダウンロード

格納装置 BOB のインタフェースボードに新しいファームウェアをダウンロードする例を次に示します (ダウンロードするファイルはデフォルトパスで指定されています)。

```
example% luxadm download -s BOB
```

例 11 SCSI 照会コマンドからの情報の表示

システム上の個々のディスクから SCSI 照会コマンドからの情報を表示する例を次に示します。サブコマンドとして一意に認識できる長さの文字列だけが使用されます。

```
example% luxadm inq /dev/rdisk/c?t?d?s2
```

例 11 SCSI 照会コマンドからの情報の表示 (続き)

例 12 ホットプラグによる取り付け

BOB,f1 という名前の格納装置の前面の最初のスロットに新しいドライブをホットプラグで取り付ける例を次に示します。

```
example% luxadm insert_device BOB,f0
```

SF880-1 という名前の Sun Fire 880 内部記憶装置サブシステムの最初のスロットに新しいドライブをホットプラグで取り付ける例を次に示します。

```
example% luxadm insert_device SF880-1,s0
```

例 13 エキスパートサブコマンドの実行

エキスパートサブコマンドを実行する例を次に示します。このサブコマンドは、格納装置 BOB のあるループを強制的に初期化します。

```
example% luxadm -e forcelp BOB
```

例 14 エキスパートモードのホットプラグサブコマンドの使用

エキスパートモードのホットプラグサブコマンドを使用して SSA 上のディスクを削除する例を以下の示します。ホットプラグ操作中の制限に関しては「注意事項」を参照してください。最初の手順では、SCSI デバイスが 2 つ目の SCSI バスによってアクセスされないように、そのデバイスを予約します。

```
example# luxadm reserve /dev/rdisk/c1t8d0s2
```

例 15 ディスクをオフラインにする

次の 2 つの手順では、ディスクをオフラインにし、バスを休止します。

```
example# luxadm -e offline /dev/rdisk/c1t8d0s2
example# luxadm -e bus_quiesce /dev/rdisk/c1t8d0s2
```

例 16 バスの休止解除

ユーザーは、この時点でディスクを取り外し、バスを休止解除し、ディスクをオンラインに戻し、ディスクの予約を解除します。

```
example# luxadm -e bus_unquiesce /dev/rdisk/c1t8d0s2
example# luxadm -e online /dev/rdisk/c1t8d0s2
example# luxadm release /dev/rdisk/c1t8d0s2
```

環境 luxadm の実行に影響のある環境変数 LANG に関しては environ(5)を参照してください。

luxadm(1M)

終了ステータス 次のいずれかの値が終了時に返されます。

0 正常終了

-1 異常終了

ファイル `usr/lib/firmware/fc_s/fc_s_fcode`
`usr/lib/firmware/ssa/ssafirmware`
`usr/lib/locale/C/LC_MESSAGES/ibfirmware`

属性 以下の属性については、`attributes(5)` を参照してください。

属性タイプ	属性値
使用条件	SUNWluxop

関連項目 `devlinks(1M)`, `disks(1M)`, `ssaadm(1M)`, `attributes(5)`, `environ(5)`, `ses(7D)`

『*Sun StorEdge A5000 設置・サービスマニュアル*』(805-4111)

注意事項 *Sun StorEdge A5000* に関するその他の情報については、『*Sun StorEdge A5000 設置・サービスマニュアル*』(805-4111) を参照してください。IEEE 拡張 WWW に関する情報については、『*Tutorial for SCSI use of IEEE company_ID*』(R. Snively 著) を参照してください。「関連項目」を参照してください。現在は、一部のデバイスドライバのみがホットプラグに対応しています。ホットプラグに対応していないディスクまたはバスにホットプラグが適用されると、以下のメッセージが表示されます。

```
luxadm: can't acquire "PATHNAME": No such file or directory
```

ルートファイルシステムまたは `/usr` ファイルシステムを含むバスやスワップデータを含むバスを休止する際は注意してください。そのようなバスを休止すると、デッドロックを引き起こす可能性があり、そのような場合は、システムの再起動が必要となります。

名前	m64config, SUNWm64_config – M64 グラフィックスアクセラレータの設定
形式	<pre> /usr/sbin/m64config [-defaults] [-depth 8 24] [-dev device-filename] [-file machine system] [-prconf] [-propt] [-res video-mode now try] [noconfirm nocheck] /usr/sbin/m64config [-prconf] [-propt] /usr/sbin/m64config [-help] [-res ?] </pre>
機能説明	<p>m64config は、M64 グラフィックスアクセラレータおよび M64 対応の X11 ウィンドウシステムのデフォルトの一部を設定します。</p> <p>形式の項に記された m64config の 1 番目の形式では、指定したオプションを OWconfig ファイルに保存します。これらのオプションは、次にウィンドウシステムをそのデバイスで実行するときに M64 デバイスを初期化するために使用されます。OWconfig ファイル内のオプションの更新は、異なるウィンドウセッションや再起動したシステムでも有効となります。</p> <p>-prconf、-propt、-help、-res ? オプションだけを起動する 2 番目と 3 番目の形式では、OWconfig ファイルは更新されません。また、3 番目の形式では、その他のオプションはすべて無視されます。</p> <p>オプションは、一度に 1 つの M64 デバイスに対してのみ指定することができます。複数の M64 デバイスに対してオプションを指定するには、m64config を複数回起動する必要があります。</p> <p>m64config で指定できるのは、M64 固有のオプションだけです。デフォルトの表示色数、デフォルトの画像表示形式クラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください (OpenWindows のマニュアルページの Xsun(1) を参照)。</p> <p>ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは、/etc/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/usr/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。</p> <p>これらの標準 OWconfig ファイルのどちらもスーパーユーザーのみが書き込みを行います。したがって、スーパーユーザーが所有する m64config プログラムは、setuid による root の権限で実行されます。</p>
オプション	<p>-defaults すべてのオプションの値をそれぞれのデフォルト値に戻します。</p> <p>-depth 8 24 表示色数を 8 または 24 ビット/ピクセルで指定します。変更を反映させるには、現在のウィンドウシステムのセッションからログアウトしてから再度ログインします。24 ビット/ピクセルでは、画面解像度を低くすることにより、ウィンドウシステムで TrueColor グラフィックスを実現できます。</p>

m64config(1M)

24 ビット/ピクセルで利用できる解像度の最大値は、PGX カードに搭載されているメモリー量に依存します。2 M バイトのメモリーを搭載した PGX カードで利用できる解像度の最大値は 800x600 です。4 M バイトのメモリーを搭載したカードで利用できる解像度の最大値は 1152x900 です。指定した解像度と色数の組み合わせに必要なメモリーが不足している場合、m64config はエラーメッセージを出力して終了します。

-dev *device-filename*

M64 特殊ファイルを指定します。デフォルトは /dev/fbs/m640 です。

-file *machine|system*

更新する OWconfig ファイルを指定します。machine を指定すると、/etc/openwin ディレクトリツリーにあるマシン固有の OWconfig ファイルが使用されます。system を指定すると、/usr/openwin ディレクトリツリーにある共通の OWconfig ファイルが使用されます。ファイルがない場合は、新たに生成されます。

-help

m64config コマンド行のオプションと機能の概要を一覧で表示します。

-prconf

M64 ハードウェア構成を表示します。次に表示例を示します。

```
--- Hardware Configuration for /dev/fbs/m640 ---
ASIC: version 0x41004754
DAC: version 0x0
PROM: version 0x0
Card possible resolutions:  640x480x60, 800x600x75, 1024x768x60
    1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76
    1280x1024x60, 1152x900x66, 1152x900x76, 1280x1024x67
    960x680x112S, 960x680x108S, 640x480x60i, 768x575x50i, 1280x800x76
    1440x900x76, 1600x1000x66, 1600x1000x76, vga, svga, 1152, 1280
    stereo, ntsc, pal
Monitor possible resolutions:  720x400x70, 720x400x88, 640x480x60
    640x480x67, 640x480x72, 640x480x75, 800x600x56, 800x600x60
    800x600x72, 800x600x75, 832x624x75, 1024x768x87, 1024x768x60
    1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76, 1152x900x66
    1152x900x76, 1280x1024x67, 960x680x112S, vga, svga, 1152, 1280
    stereo
Possible depths: 8, 24
Current resolution setting: 1280x1024x76
Current depth: 8
```

-propt

-file オプションで指定された OWconfig ファイルに書かれた M64 オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。m64config の呼び出しが終了した後に、OWconfig ファイルに書き込まれるオプションの値を表示します。次に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/m640 ---
OWconfig: machine
Video Mode: not set
```

```
-res video-mode [ now | try [ noconfirm | nocheck ] ]
```

指定した M64 デバイ스에 接続されている 모니터를 制御する 際 に 使われる 表示 모드 を 指定します。表示モードはあらかじめ組み込まれています。表示モードの形式は *widthx-heightxrate* で、*width* はピクセル単位の画面幅、*height* はピクセル単位の画面の高さ、*rate* は画面を垂直方向に再描画する周期です。便宜上、*-res* にリフレッシュレートを指定する際は、値の直前に *x* の代わりに *@* を使用することができます。たとえば、1280x1024@76 のように指定することができます。

有効な表示モードのリストは、`m64config -res '?'` を実行することによって得ることができます。? は引用符で囲んで文字として扱う必要があります。すべてのビデオボードおよびモニターがすべての解像度に対応しているわけではありません。m64config は、ボードが対応していない解像度に設定することを許しません。モニターが対応していない解像度に設定しようとすると、確認を促すメッセージを表示します。

記号名

便宜上、上記の表示モードのいくつかには記号名が定義されています。*widthxheightxrate* の形式の代わりに、記号名を *-res* の引数として指定することができます。記号名 *none* は、ウィンドウシステムを実行すると、画面の解像度は現在デバイスにプログラムされている表示モードになることを意味します。

記号名	対応する表示モード
svga	1024x768x60
1152	1152x900x76
1280	1280x1024x76
none	(現在デバイスにプログラムされている表示モード)

-res オプションには、表示モードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定することができます。

nocheck

このオプションを指定すると、モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定された表示モードは、現在接続されているモニターに適切かどうかにかかわらず受け付けられます。このオプションは、M64 デバイ스에 異なる 모니터를 接続する場合に便利です。このオプションを指定すると、*noconfirm* も指定されます。

noconfirm

-res オプションを指定した際に、システムが使用可能であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために m64config のデフォルトの動作では、この問題についての警告メッセージと、処理を継続するかどうかを確認するメッセージを表示します。*noconfirm* オプションを指定すると、m64config コマンドはこの確認をせずに、

m64config(1M)

要求のあった表示モードにプログラムします。このオプションは、m64config がシェルスクリプトから実行されている場合に便利です。

now

OWconfig ファイルの表示モードを更新するとともに、M64 デバイスが指定した表示モードにただちにプログラムされます (この機能は、ウィンドウシステムを開始する前に表示モードを変更する際に便利です)。

対象となるデバイスが稼働している間 (たとえば、ウィンドウシステムの稼働中) に、この追加オプションを m64config に指定することはお勧めしません。予期しない結果になることもあります。now オプションを指定して m64config コマンドを実行する場合は、最初にウィンドウシステムを終了してください。now オプションがウィンドウシステムのセッション中に使用された場合、表示モードはただちに変更されますが、画面の幅や高さはそのセッションが終了して次のセッションに入るまで変更されません。さらに、立体表示モードではシステムが変更を認識しないことがあります。したがって、ウィンドウシステムの稼働中には絶対に now オプションを指定しないでください。

try

このオプションを指定すると、指定した表示モードが試験的にプログラムされます。ユーザーは、指定した表示モードを使用する場合は、メッセージが表示されてから 10 秒以内に y と入力します。表示されたモードを使用しない場合は、10 秒以内に任意の文字を入力します。y または Return キー以外の文字の入力は、すべて「使用しない」とみなされ、以前の表示モードに戻され、OWconfig ファイル中の表示モードは書き換えられません (その他の指定されたオプションは有効となります)。Return キーの入力があつた場合は、新しい表示モードを保持するかどうかを yes または no で確認するメッセージが表示されます。このオプションを指定すると、now サブオプションも指定されます。(now サブオプションの注意を参照)。

デフォルト設定

m64config コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。

ウィンドウシステムを実行する際に、m64config による M64 オプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルトを以下に示します。

オプション	デフォルト値
-dev	/dev/fbs/m640
-file	machine
-res	none

-res オプションのデフォルト値 none とは、ウィンドウシステムが実行された場合に、画面解像度がそのデバイスに現在プログラムされている表示モードになることを意味しています。

m64config(1M)

これによって、PROM によってデバイスの解像度を指定しているユーザーとの共用性が保てます。(GX などの)一部のデバイスでは、PROM が表示モードを指定する唯一の手段です。これは、デフォルトの M64 表示モードは、最終的に PROM によって決まることを意味しています。

使用例 例 1 モニターの種類の変更

モニターの種類を、垂直周波数 76 Hz で解像度 1280x1024 に変更する例を以下に示します。

```
example% /usr/sbin/m64config -res 1280x1024x76
```

ファイル /dev/fbs/m640
デバイス特殊ファイル

/usr/openwin/server/etc/OWconfig
システム設定ファイル

/usr/lib/fbconfig/SUNWm64_config
usr/sbin/m64config へのシンボリックリンク

属性 以下の属性については、attributes(5)を参照してください。

属性タイプ	属性値
使用条件	SUNWm64cf

関連項目 attributes(5), m64(7D)

『OpenWindows デスクトップ・リファレンスマニュアル』

modinfo(1M)

名前	modinfo - ロードされているカーネルモジュールについての情報の表示				
形式	<code>/usr/sbin/modinfo [-c] [-w] [-i module-id]</code>				
機能説明	<p>modinfo ユーティリティは、ロードされているモジュールに関する情報を表示します。表示形式は次のとおりです。</p> <pre>Id Loadaddr Size Info Rev Module Name</pre> <p><i>Id</i> はモジュール ID、<i>Loadaddr</i> はテキスト開始位置のアドレス (16 進表記)、<i>Size</i> はテキストとデータと <i>bss</i> のバイト単位の合計サイズ (16 進表記)、<i>Info</i> はモジュール固有の情報、<i>Rev</i> はロード可能モジュールシステムのリビジョン番号、そして <i>Module Name</i> はモジュールのファイル名と内容の説明を表します。</p> <p>モジュール固有の情報として表示される内容は、モジュールの種類により異なります。すなわち、ドライバであればブロックメジャー番号とキャラクタメジャー番号、システムコールであればシステムコール番号、その他の種類であれば以下に示すようなカーネルテーブルへのインデックスが表示されます。</p> <p>fmodsw STREAMS モジュールの場合 vfssw ファイルシステムの場合 class スケジューリングクラスの場合 execsw exec モジュールの場合</p>				
オプション	<p>次のオプションを指定できます。</p> <p>-c ロードされているモジュールのインスタンス数と、モジュールの現在の状態を表示します。</p> <p>-i <i>module-id</i> 指定したモジュールに関する情報のみを表示します。</p> <p>-w モジュールに関する情報を、80 文字の位置で切り落としません。</p>				
使用例	<p>例 1 モジュールに関する情報の表示</p> <p>次の例では、モジュール 3 に関する情報を表示しています。</p> <pre>example% modinfo -i 3 Id Loadaddr Size Info Rev Module Name 3 f5a7a000 3bc0 1 1 spedfs (specfs のファイルシステム)</pre>				
属性	<p>次の属性については attributes(5) のマニュアルページを参照してください。</p> <table border="1"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWcsu</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWcsu
属性タイプ	属性値				
使用条件	SUNWcsu				
関連項目	modload(1M), modunload(1M), attributes(5)				

名前	modload – カーネルモジュールのロード				
形式	modload [-p] [-e <i>exec_file</i>] <i>filename</i>				
機能説明	<p>modload は、<i>filename</i> で示すロード可能モジュールを、稼動中のシステムにロードします。<i>filename</i> は、<code>ld -r</code>で生成されたオブジェクトファイルを示します。<i>filename</i> が絶対パス名の場合、その絶対パスが指定するファイルがロードされます。<i>filename</i> の先頭文字がスラッシュ (/) ではない場合、<code>-p</code> オプションが指定されていないと、現在のディレクトリに相対するパスを使って <i>filename</i> をロードします。カーネルのモジュールパス (<code>modpath</code>) 変数は、<code>/etc/system</code> ファイルを使って設定できます。この変数のデフォルト値は、オペレーティングシステムをロードした時のパスです。通常は、<code>/kernel /usr/kernel</code> となります。したがって、次のように入力した場合、カーネルは <code>./drv/foo</code> ファイルを探します。</p> <pre>example# modload drv/foo</pre> <p>また、次のように入力した場合には、カーネルはまず <code>/kernel/drv/foo</code> を探し、存在しなければ <code>/usr/kernel/drv/foo</code> を探します。</p> <pre>example# modload -p drv/foo</pre>				
オプション	<p><code>-p</code> モジュールを検索するパスとして、カーネルの内部 <code>modpath</code> 変数を使用します。</p> <p><code>-e <i>exec_file</i></code> モジュールのロードが正常に終了した後で実行すべき シェルスクリプトまたは実行可能イメージの名前を指定します。そのスクリプトまたはイメージにはいくつかの引数が渡されます。第 1 引数は常にモジュール ID (10 進数) です。他の引数は、モジュールの種類により異なります。ドライバには、ブロックメジャー番号とキャラクタメジャー番号、システムコールにはシステムコール番号、その他のモジュールタイプにはそれぞれにあったカーネルテーブルへのインデックスが指定されます (<code>modinfo(1M)</code> を参照)。</p>				
属性	<p>次の属性については <code>attributes(5)</code> のマニュアルページを参照してください。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">属性タイプ</th> <th style="text-align: center;">属性値</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">使用条件</td> <td style="text-align: center;">SUNWcsu</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWcsu
属性タイプ	属性値				
使用条件	SUNWcsu				
関連項目	<p><code>ld(1)</code>, <code>add_drv(1M)</code>, <code>kernel(1M)</code>, <code>modinfo(1M)</code>, <code>modunload(1M)</code>, <code>system(4)</code>, <code>attributes(5)</code>, <code>modldrv(9S)</code>, <code>modlinkage(9S)</code>, <code>modlstrmod(9S)</code>, <code>module_info(9S)</code></p> <p><i>Writing Device Drivers</i></p> <p><i>Solaris 移行ガイド</i></p>				

modload(1M)

注意事項 デバイスドライバを追加するには、modloadではなく add_drv(1M) を使います。デバイスドライバの追加方法については、*Writing Device Drivers* を参照してください。

名前	modunload – モジュールのアンロード				
形式	modunload -i <i>module_id</i> [-exec <i>exec_file</i>]				
機能説明	modunload は、稼働中のシステムからロード可能モジュールをアンロードします。 <i>module_id</i> はアンロードするモジュールの ID で、これは modinfo(1M) の出力情報中に得られる値と同じです。ID として 0 を指定した場合、自動ローディングされたモジュールのうちアンロード可能なものがすべてアンロードされます。modload(1M) を使ってロードしたモジュールは対象とはなりません。				
オプション	<p>-i <i>module_id</i> アンロードすべきモジュールを指定します。</p> <p>-e <i>exec_file</i> モジュールをアンロードする前に実行すべきシェルスクリプトまたは実行可能イメージの名前を指定します。そのスクリプトまたはイメージにはいくつかの引数が渡されます。第 1 引数は常にモジュール ID (10 進数) です。他の 2 つの引数は、モジュールの種類により異なります。ロード可能ドライバの場合は、ブロックメジャー番号が第 2 引数、キャラクタメジャー番号が第 3 引数です。ロード可能システムコールの場合は、システムコール番号が第 2 引数です。ロード可能 exec クラスの場合は、execsw テーブルへのインデックスが第 2 引数です。ロード可能ファイルシステムの場合は、vfssw テーブルへのインデックスが第 2 引数です。ロード可能ストリームモジュールの場合は、fmodsw テーブルへのインデックスが第 2 引数です。ロード可能スケジューリングクラスの場合は、クラス配列へのインデックスが第 2 引数です。該当しない引数の値としては、-1 が渡されます。</p>				
属性	次の属性については attributes(5) のマニュアルページを参照してください。				
	<table border="1"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWcsu</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWcsu
属性タイプ	属性値				
使用条件	SUNWcsu				
関連項目	modinfo(1M), modload(1M), attributes(5)				

patchadd(1M)

名前	patchadd – Solaris 2.x および Solaris 7 システムへのパッチ適用
形式	<pre>patchadd [-d] [-u] [-B backout_dir] [-C net_install_image -R client_root_path -S service] patch</pre> <pre>patchadd [-d] [-u] [-B backout_dir] [-C net_install_image -R client_root_path -S service] -M patch_dir patch_id... patch_dirpatch_list</pre> <pre>patchadd [-C net_install_image -R client_root_path -S service] -p</pre>
機能説明	<p>patchadd は、Solaris 2.x およびその互換バージョンのシステムにパッチを適用しません。patchadd は、Solaris 1.x システム用のパッチを適用するためには使用できません。patchadd を実行するにはスーパーユーザーになる必要があります。</p> <p>patchadd コマンドには、次の3つの使用方法があります。</p> <ol style="list-style-type: none">(1) 1つのパッチを、1つのシステム、クライアント、サービス、またはネットインストールイメージの miniroot にインストールします。(2) 複数のパッチを、1つのシステム、クライアント、サービス、またはネットインストールイメージの miniroot にインストールします。(3) クライアント、サービス、またはネットインストールイメージの miniroot にインストールされたパッチを表示します。
オプション	<p>以下のオプションを指定できます。</p> <p>-d パッチが適用されるファイルのバックアップを作成しません。このオプションを指定すると、適用されたパッチを後で削除 (バックアウト) することはできません。</p> <p>-p 現在適用されているパッチのリストを表示します。</p> <p>-u パッチ適用前にファイルを検証せずに、パッチを無条件に適用します。パッチの適用によって変更されるファイルがある場合でも、パッチを適用します。</p> <p>-B backout_dir パッチのバックアウト (削除) 時に利用されるデータ (バックアウトデータ) を、パッケージデータベース以外のディレクトリに保存します。backout_dir は絶対パス名で指定してください。</p> <p>-C net_install_image setup_install_server によって作成された、ネットインストールイメージ上の miniroot にあるファイルに対してパッチを適用します。net_install_image には、Solaris 2.6 およびその互換バージョンのブートディレクトリへの絶対パス名を指定してください (「使用例」の項を参照)。</p> <p>-M patch_dir patch_id ... patch_dir patch_list インストールするパッチを指定します。次の指定方法があります。</p> <ol style="list-style-type: none">1. ディレクトリ名とパッチ番号を指定

パッチが置かれているディレクトリを *patch_dir* に絶対パス名で指定します。インストールするパッチの番号を *patch_id* に指定します。複数の *patch_id* を指定することをお勧めします。

2. ディレクトリ名と、パッチのリストが記述されたファイルの名前を指定

インストールするパッチがリストされているファイルが置かれているディレクトリの名前を *patch_dir* に絶対パス名で指定します。インストールするパッチがリストされているファイルの名前を *patch_list* に指定します。

-R *client_root_path*

patchadd によって生成されるすべてのパッチファイルを、ディレクトリ *client_root_path* の下に置きます。*client_root_path* には、サーバー上のクライアントのルートディレクトリを絶対パス名で指定します。この *-R* オプションは *-S* オプションと同時に指定することはできません。詳細は「注意事項」の項を参照してください。

-S *service*

代替サービス (たとえば *Solaris_2.3*) を指定します。ここでのサービスとは、クライアント・サーバーモデルの OS サービスのことを指しており、サーバーのコンソールでのみ使用できます。サーバーは、ホストマネージャで作成された */usr* 共有ファイルシステムを持つことができ、登録された OS サービスをクライアントが利用できるようにすることが可能です。この *-s* オプションは *-R* オプションと同時に指定することはできません。詳細は「注意事項」の項を参照してください。

オペランド 以下のオペランドを指定できます。

<i>patch</i>	<i>patch_id</i> への絶対パス名。 <i>patch</i> にはたとえば <i>/var/sadm/spool/patch/104945-02</i> のように指定します。
<i>patch_dir</i>	パッチが置かれているディレクトリへの絶対パス名。たとえば <i>/var/sadm/spool/patch</i> のように指定します。
<i>patch_id</i>	パッチ番号。たとえば <i>104945-02</i> のように指定します。
<i>patch_list</i>	インストールするパッチのリストが記述されたファイルの名前。ファイル <i>patch_list</i> には、1 行に 1 つの <i>patch_id</i> を記述します。

使用例 以下に示す例では、*/usr/sbin* ディレクトリのコマンドを使用しているものとします。

例 1 スタンドアロンマシンに 1 つのパッチをインストールする

スタンドアロンマシンに 1 つのパッチをインストールする例を示します。

```
example# patchadd /var/spool/patch/104945-02
```

例 2 サーバーのコンソールからクライアントに 1 つのパッチをインストールする

サーバーのコンソールからクライアントに 1 つのパッチをインストールする例を示します。

```
example# patchadd -R /export/root/client1 /var/spool/patch/104945-02
```

patchadd(1M)

例 2 サーバーのコンソールからクライアントに1つのパッチをインストールする (続き)

例 3 サーバーのコンソールから OS サービスに1つのパッチをインストールする

サーバーのコンソールからクライアントに1つのパッチをインストールする例を示します。

```
example# patchadd -S Solaris_2.3 /var/spool/patch/104945-02
```

例 4 patchadd を1回実行して複数のパッチをインストールする

1回の patchadd の実行で複数のパッチをインストールする例を示します。

```
example# patchadd -M /var/spool/patch 104945-02 104946-02 102345-02
```

例 5 パッチのリストが記述されているファイルを指定して複数のパッチをインストールする

インストールするパッチのリストが記述されたファイルを指定して、複数のパッチをインストールする例を示します。

```
example# patchadd -M /var/spool/patch patchlist
```

例 6 クライアントに複数のパッチをインストールし、バックアウトデータをデフォルト以外のディレクトリに保存する

クライアントに複数のパッチをインストールし、パッチのバックアウト時に利用されるデータ (バックアウトデータ) をデフォルト以外のディレクトリに保存する例を示します。

```
example# patchadd -M /var/spool/patch -R /export/root/client1 -B /export/backoutrepository 104945-02
```

例 7 Solaris 2.6 およびその互換バージョンのネットインストールイメージにパッチをインストールする

Solaris 2.6 およびその互換バージョンのネットインストールイメージにパッチをインストールする例を示します。

```
example# patchadd -C /export/Solaris_2.6/Tools/Boot /var/spool/patch/104945-02
```

例 8 クライアントにインストールされているパッチを表示する

クライアント上にインストールされているパッチを表示する例を示します。

```
example# patchadd -R /export/root/client1 -p
```

終了ステータス 以下の終了ステータスが返されます。

0 正常終了

>0 エラーが発生した

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWswmt, SUNWcsu

診断
パッチのインストールエラー

パッチのインストール時によく発生する問題、出力されるエラーメッセージ、その対処方法について説明します。

メッセージ

```
The prepatch script exited with return code retcode.
patchadd is terminating.
```

説明・対処法

パッチに付属している `prepatch` スクリプトが 0 以外の終了コードで終了しました。 `prepatch` スクリプトのトレースを実行して、 `prepatch` スクリプトが不正な終了コードで終了した原因を調べてください。問題を修正するには、 `prepatch` スクリプトの先頭行に `-x` オプションを追加して再度 `patchadd` を実行してください。

メッセージ

```
The postpatch script exited with return code retcode.
Backing out patch.
```

説明・対処方法

パッチに付属している `postpatch` スクリプトが 0 以外の終了コードで終了しました。このスクリプトはおもに、パッチパッケージのオブジェクトに対応しないファイルを一掃するため (つまりパッケージの所有権およびアクセス権に問題があるとき) に使用されます。出力された検証エラーをすべて確認し、それぞれについて適切な処置を行なった後、 `-u` オプション付きで再度 `patchadd` を実行してください。検証エラーを無視してパッチがインストールされます。

メッセージ

```
Insufficient space in /var/sadm/patch to save old files.
(Solaris 2.4 およびそれ以前のシステム)
```

説明・対処方法

パッチ適用前のファイルを保存するための容量が `/var/sadm/patch` ディレクトリにありません。3つの対処法があります。(1) `-B` オプション付きで `patchadd` を実行して、指定したファイルシステムにバックアウトデータ (パッチのバックアウト時に利用されるデータ) を保存します。(2) 不要なファイルを削除することによってディスク容量を確保します。(3) `-d` オプション付きで `patchadd` を実行して、パッチ適用前のファイルを保存しないようにします。

パッチ適用前のファイルを保存しないように選択した場合、後に `patchrm` を使用してパッチを削除することはできなくなります。以前に適用したパッチの保存領域を削除することによって、システム領域を確保するという方法もあります。後にパッチを削除する可能性はないと判断した場合は、 `patchadd` によって保存されたファイルを削除しても構いません。パッチ `patch_id` について保存されたファイルを削除するには、次のように実行してください。

patchadd(1M)

```
cd /var/sadm/patch/patch_id
rm -r save/*
rm .oldfilessaved
```

上記のコマンドを実行後は、パッチ *patch_id* は削除できなくなります。

メッセージ

```
Insufficient space in /var/sadm/pkg/PKG/save to save old files.
(Solaris 2.5 およびそれ以降のシステム)
```

説明・対処方法

/var/sadm/pkg/PKG/save ディレクトリに容量が不足しているため、パッチ適用前のファイルを保存できません。次の3つの対処方法があります。(1) -B オプション付きで patchadd を実行して、指定したファイルシステムにバックアウトデータ (パッチのバックアウト時に利用されるデータ) を保存します (1つ前のメッセージに関する説明を参照してください)。(2) 不要なファイルを削除することによってディスク領域を確保します。(3) -d オプション付きで patchadd を実行して、パッチ適用前のファイルを保存しないようにします。

パッチ適用前のファイルを保存しないように選択した場合、後に patchrm を使用してパッチを削除することはできなくなります。以前に適用したパッチの保存領域を削除することによって、システム領域を確保するという方法もあります。後にパッチを削除する可能性がないと判断した場合は、patchadd によって保存されたファイルを削除しても構いません。patch_id について保存されたファイルを削除するには、次のように実行してください。

```
cd /var/sadm/pkg/pkgabbrev/save
rm -r patch_id
```

上記のコマンドを実行後は、パッチ *patch_id* は削除できなくなります。

メッセージ

```
Save of old files failed.
(Solaris 2.4 およびそれ以前のシステム)
```

説明・対処方法

パッチを適用する前に、パッチインストールスクリプトは cpio を使用してパッチ適用前のファイルを保存します。このエラーメッセージは、cpio が失敗したことを示します。このエラーメッセージの前に cpio からの出力が表示されているはずですが、cpio の失敗を修正するために適切な処置を行う必要があります。次の2つの対処方法があります。(1) 不要なファイルを削除してディスク領域を確保します。(2) -d オプション付きで patchadd を実行してパッチ適用前のファイルを保存しないようにします。ただし、パッチ適用前のファイルを保存しないように選択した場合、パッチを削除することはできなくなります。

メッセージ

```
Pkgadd of pkgname package failed with error code code.
See /tmp/log.patch_id for reason for failure.
```

説明・対処方法

メッセージ中に示されたパッチパッケージのインストールに失敗しました。patchadd はパッチをバックアウトして、システムをパッチ適用前の状態にしま

す。ログファイルを参照してインストールに失敗した原因を確認し、必要な処置を行なって問題を解決した後、再度パッチを適用してください。

メッセージ

```
Pkgadd of pkgname package failed with error code code.
Will not backout patch...patch re-installation.
Warning: The system may be in an unstable state!
See /tmp/log.patch_id for reason for failure.
```

説明・対処方法

メッセージ中に示されたパッチパッケージのインストールに失敗しました。patchadd はパッチをバックアウトしません。ログファイルを参照して pkgadd が失敗した原因を確認し、必要な処置を行なって問題を解決してください。その後、patchrm を使用して手動でパッチをバックアウトしてから、再度パッチを適用してください。

メッセージ

```
patchadd is unable to find the INST_RELEASE file. This file
must be present for patchadd to function correctly.
```

説明・対処方法

システムに INST_RELEASE ファイルがありません。このファイルは初期インストール中またはアップグレード中に作成されます。

メッセージ

```
A previous installation of patch patch_id was invoked
that saved files that were to be patched. Since files
were saved, you must run this instance of patchadd
without the -d option.
```

説明・対処方法

以前に -d オプションを使用しないでパッチがインストールされている場合、パッチを再インストールするときにも -d オプションなしで実行する必要があります。-d オプションを付けずに patchadd を実行してください。

メッセージ

```
A previous installation of patch patch_id was invoked
with the -d option. (i.e. Do not save files that would
be patched) Therefore, this invocation of patchadd
must also be run with the -d option.
```

説明・対処方法

以前に -d オプションを使用してパッチがインストールされている場合、パッチを再インストールするときにも -d オプションを使用する必要があります。-d オプション付きで patchadd を実行してください。

その他の診断 メッセージ

以下に示すパッチインストール時のメッセージは、「説明・対処方法」で説明しているように必ずしもエラーではありませんが、パッチインストールのログファイルに記録されます。

patchadd(1M)

メッセージ

```
Package not patched:  
PKG=SUNxxxx  
Original package not installed
```

説明・対処方法

メッセージに示されているパッチコンポーネントは、システムにインストールされていないパッケージに対するパッチです。これは必ずしもエラーではありません。1つのパッチが1つのバグを複数のパッケージに対して修正することもあります。

たとえば、オンラインバックアップと `fddi` パッケージの両方に対する1つのバグを修正するパッチを例として考えます。オンラインバックアップはインストールされているけれども `fddi` パッケージがインストールされていない場合、次のようなメッセージが出力されます。:

```
Package not patched:  
PKG=SUNWbf  
Original package not installed
```

システム上に `fddi` パッケージがインストールされている場合には、このメッセージをエラーとして対処する必要があります。必要な処置を行なってパッケージをインストールし、(パッチによって他のパッケージがインストールされている場合は)パッチをバックアウトし、再度パッチをインストールしてください。

メッセージ

```
Package not patched:  
PKG=SUNxxx  
ARCH=xxxxxxxx  
VERSION=xxxxxxxx  
Architecture mismatch
```

説明・対処方法

メッセージに示されているパッチコンポーネントは、ユーザーが使用しているシステムとは異なるアーキテクチャのパッケージに対するパッチです。これは必ずしもエラーではありません。アーキテクチャ固有のパッケージに対するパッチには、該当する各アーキテクチャごとに1つのコンポーネントが含まれている場合もあります。たとえば `sun4m` アーキテクチャのシステムを使用している場合に `SUNWcar` パッケージに対するパッチをインストールしようとする、次のようなメッセージが出力されます。

```
Package not patched:  
PKG=SUNWcar  
ARCH=sparc.sun4c  
VERSION=11.5.0,REV=2.0.18  
Architecture mismatch
```

```
Package not patched:  
PKG=SUNWcar  
ARCH=sparc.sun4d  
VERSION=11.5.0,REV=2.0.18  
Architecture mismatch
```

```
Package not patched:
PKG=SUNWcar
ARCH=sparc.sun4e
VERSION=11.5.0,REV=2.0.18
Architecture mismatch
```

```
Package not patched:
PKG=SUNWcar
ARCH=sparc.sun4u
VERSION=11.5.0,REV=2.0.18
Architecture mismatch
```

使用しているシステムのアーキテクチャを patchadd が正しく認識していない場合のみ、このメッセージはエラーを示しています。

メッセージ

```
Package not patched:
PKG=SUNxxxx
ARCH=xxxx
VERSION=xxxxxxxx
Version mismatch
```

説明・対処方法

パッチが適用されるソフトウェアバージョンがシステムにインストールされていません。たとえば SunOS 5.5 を実行している場合に SunOS 5.6 に対するパッチをインストールしようとする、次のようなメッセージが出力されます。

```
Package not patched:
PKG=SUNWcsu
ARCH=sparc
VERSION=10.0.2
Version mismatch
```

これは必ずしもエラーではありません。パッチを適用するパッケージのバージョンが不一致の場合は、正しいバージョンのパッチを入手するか、または正しいバージョンのパッケージをインストールしてください。そのあと、必要な場合はパッチをバックアウトしてから、再度パッチを適用してください。

メッセージ

```
Re-installing Patch.
```

説明・対処方法

インストールしようとしているパッチはすでに適用されていますが、追加インストールされるパッケージが少なくとも 1 つパッチに含まれています。たとえば、AnswerBook がインストールされていないシステムに、OpenWindows と AnswerBook の両方のコンポーネントが含まれているパッチを適用すると、そのパッチの AnswerBook コンポーネント部分は適用されません。後に pkgadd を使って AnswerBook をインストールしてパッチを再度適用すると、そのパッチの AnswerBook コンポーネント部分がシステムに適用されます。

メッセージ

```
patchadd Interrupted.
patchadd is terminating.
```

patchadd(1M)

	<p>説明・対処方法</p> <p>patchadd の実行が中断されました (通常 CTRL-c が押されたため)。patchadd は処理中のファイルを削除し、終了します。</p>
	<p>メッセージ</p> <pre>patchadd Interrupted. Backing out Patch...</pre>
	<p>説明・対処方法</p> <p>patchadd の実行が中断されました (通常 CTRL-c が押されたため)。patchadd は処理中のファイルを削除し、パッチをバックアウトし、終了します。</p>
関連項目	cpio(1), pkginfo(1), patchrm(1M), pkgadd(1M), pkgchk(1M), pkgrm(1M), showrev(1M), attributes(5)
注意事項	<p>クライアントまたはサーバーにパッチをインストールするには、patchadd を 2 回実行する必要があります。1 回は -R オプションを付けて実行し、もう 1 回は -s オプションを付けて実行します。これによって、パッチが /usr と / (ルート) パーティションの両方に確実にインストールされます。パッチに /usr と / (ルート) のパッケージが含まれている場合に、上記の方法を実行する必要があります。</p> <p>patchadd を実行すると、pkgadd が起動され、<i>pkg/install</i> ディレクトリにあるインストールスクリプトを実行します。checkinstall スクリプトは、所有権を <i>install</i> というユーザー名に設定して実行されます。ユーザー名が <i>install</i> に設定されていない場合、pkgadd は checkinstall スクリプトを <i>nobody</i> として実行します。SVR4 ABI には、checkinstall は情報収集を行うスクリプトとしてのみ使用されると記述されています。checkinstall スクリプトのアクセス権を初期設定から変更すると、pkgadd はファイルをオープンして読み取ることができなくなり、次のようなメッセージが出力されてパッチのインストールが異常終了します。</p> <pre>pkgadd: ERROR: checkinstall script did not complete successfully.</pre> <p>このため、checkinstall スクリプトのアクセス権は変更しないでください。パッチが正しくインストールされたときのパッチインストールログファイルの内容は、patchadd が pkgadd の出力をリダイレクトしたものになります。パッチが正しくインストールされると、pkgadd は次のようなメッセージを出力し、それがログファイルに書き込まれます。</p> <pre>This appears to be an attempt to install the same architecture and version of a package which is already installed. This installation will attempt to overwrite this package.</pre> <p>このメッセージは処理の失敗を示しているものではありません。パッチが正しくインストールされたときの pkgadd による正しい動作を示しています。</p> <p>クライアント・サーバーマシンでは、既存のクライアントにまたはクライアントのルートディレクトリ (<i>templates</i> 領域) にパッチパッケージは適用 (インストール) されません。このため、すべてのクライアントマシンにおいて直接 patchadd を使用して、適切な時にクライアントマシンにパッチを適用する必要があります。パッケージユーティリティ (pkgadd, pkgrm, pkgchk) に影響するバグは、patchadd または patchrm の動作に影響する場合があります。patchadd および patchrm は、上記の</p>

パッケージユーティリティを使用してパッチパッケージをインストールおよびバックアウトしています。これらのパッケージユーティリティのバグを修正するパッチが提供されているかどうかを確認し、提供されている場合はそのパッチを先に適用してから、他のパッチを適用することをお勧めします。現在提供されている、パッケージユーティリティに対するパッチは、次のとおりです。

Solaris 2.1:

100901

Solaris 2.2:

101122

Solaris 2.3:

101331

Solaris 2.4 (SPARC 版):

102039

Solaris 2.4 (Intel 版):

102041

Solaris 2.5.1 (SPARC 版):

104578

Solaris 2.5.1 (Intel 版):

104579

Solaris 2.6 (SPARC 版):

106292

Solaris 2.6 (Intel 版):

106293

patchrm(1M)

名前	patchrm – Solaris 2.x および Solaris 7 システムからのパッチ削除とパッチ適用前のファイル復元
形式	patchrm [-f] [-B <i>backout_dir</i>] [-C <i>net_install_image</i> -R <i>client_root_path</i> -S <i>service</i>] <i>patch_id</i>
機能説明	patchrm は、パッチパッケージを削除 (バックアウト) し、Solaris 2.x または Solaris 7 システムに保存されているパッチ適用前のファイルを復元します。patchrm は Solaris 1.x システム用のパッチには使用できません。patchrm を実行するにはスーパーユーザーになる必要があります。
オプション	以下のオプションを指定できます。 -f 他のパッチに置き換えられているかどうかに関係なく、指定したパッチを削除 (バックアウト) します。 -B <i>backout_dir</i> パッケージデータベースとは別のディレクトリにバックアウトデータ (パッチのバックアウト時に利用されるデータ) が保存されているパッチを、バックアウトします。パッチのインストール時に patchadd コマンドでデフォルトのバックアウトデータのディレクトリを変更した場合のみ、このオプションを使用します。 <i>backout_dir</i> には絶対パス名を指定してください。 -C <i>net_install_image</i> setup_install_server によって作成された、ネットインストールイメージ上の miniroot にあるファイルに対して適用されたパッチを削除 (バックアウト) します。 <i>net_install_image</i> には、Solaris 2.6 およびその互換バージョンのブートディレクトリへの絶対パス名を指定してください。 -R <i>client_root_path</i> patchrm によって生成されるすべてのパッチファイルをディレクトリ <i>client_root_path</i> の下に置きます。 <i>client_root_path</i> には、サーバー上のクライアントのルートディレクトリを絶対パス名で指定します。この -R オプションは -S オプションと同時に指定することはできません。 -S <i>service</i> 代替サービス (たとえば Solaris_2.3) を指定します。ここでのサービスとは、クライアント・サーバーモデルの OS サービスのことを指しており、サーバーのコンソールでのみ使用できます。サーバーは、ホストマネージャで作成された /usr 共有ファイルシステムを持つことができ、登録された OS サービスをクライアントが利用できるようにすることが可能です。この -S オプションは -R オプションと同時に指定することはできません。
オペランド	以下のオペランドを指定できます。 <i>patch_id</i> パッチ番号。たとえば 104945-02 のように指定します。

使用例 以下に示す例では、パッチ 104945-02 がシステムにインストールされていることを前提としています。また、/usr/sbin ディレクトリのコマンドを使用しているものとします。

例 1 スタンドアロンシステム上の 1 つのパッチを削除する

スタンドアロンシステム上のパッチを 1 つ削除 (バックアウト) する例を示します。

```
example# patchrm 104945-02
```

例 2 サーバーのコンソールからクライアントシステム上の 1 つのパッチを削除する

サーバーのコンソールから、クライアントシステム上のパッチを 1 つ削除 (バックアウト) する例を示します。

```
example# patchrm -R /export/root/client1 104945-02
```

例 3 サーバーの OS サービス領域上の 1 つのパッチを削除する

サーバーの OS サービス領域にあるパッチを 1 つ削除 (バックアウト) する例を示します。

```
example# patchrm -S Solaris_2.3 104945-02
```

例 4 ネットインストールイメージ上の 1 つのパッチを削除する

ネットインストールイメージ上のパッチを 1 つ削除 (バックアウト) する例を示します。

```
example# patchrm -C /export/Solaris_2.6/Tools/Boot 104945-02
```

終了ステータス 以下の終了ステータスが返されます。

```
0          正常終了
>0        エラーが発生した
```

属性 次の属性については `attributes(5)` のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWswmt, SUNWcsu

診断 パッチのバックアウト時によく発生する問題、出力されるエラーメッセージ、その対処方法について説明します。

メッセージ

```
prebackout patch exited with return code code.
patchrm exiting.
```

patchrm(1M)

説明・対処方法

パッチに付属している `prebackout` スクリプトが 0 以外の終了コードで終了しました。 `prebackout` スクリプトのトレースを作成して、 `prebackout` スクリプトが失敗した原因を調べてください。問題を修正するには、 `prebackout` スクリプトの先頭行に `-x` オプションを追加して再度 `patchrm` を実行してください。

メッセージ

```
postbackout patch exited with return code code.
patchrm exiting.
```

説明・対処方法

パッチに付属している `postbackout` スクリプトが 0 以外の終了コードで終了しました。 `postbackout` スクリプトを参照して `postbackout` スクリプトが失敗した原因を調べてください。問題を修正するには、 `postbackout` スクリプトの先頭行に `-x` オプションを追加して、必要な場合は `postbackout` スクリプトだけを再度実行してください。

メッセージ

```
Only one service may be defined.
```

説明・対処方法

複数の OS サービスからパッチをバックアウトしようとしています。複数の OS サービスからパッチをバックアウトするには、各 OS サービスごとに別々に `patchrm` を実行してください。

メッセージ

```
The -S and -R arguments are mutually exclusive.
```

説明・対処方法

ネイティブでない OS サービスと `client_root_path` からパッチをバックアウトしようとしています。これら 2 つの引数は互いに排他的です。ネイティブでない `usr` パーティションからパッチをバックアウト (削除) するには `-s` オプションを使用する必要があります。クライアントの (ネイティブまたは非ネイティブの) ルートパーティションからパッチをバックアウトするには `-R` オプションを使用する必要があります。

メッセージ

```
The service service cannot be found on this system
```

説明・対処方法

ネイティブでない OS サービスからパッチをバックアウト (削除) しようとしたが、指定された OS サービスはシステムにインストールされていません。正しい OS サービスを指定してください。

メッセージ

```
Only one client_root_path may be defined.
```

説明・対処方法

`-R` オプションを指定して複数の `client_root_path` を指定しています。1 回の `patchrm` の実行につき `-R` オプションは 1 回だけ使用できます。

メッセージ

The *dir* directory cannot be found on this system.

説明・対処方法

-R オプションを使用して、マウントされていないまたはシステムに存在しないディレクトリを指定しています。正しいディレクトリ名を指定して、パッチのバックアウトを再度実行してください。

メッセージ

Patch *patch_id* has not been successfully installed to this system.

説明・対処方法

システムにインストールされていないパッチをバックアウト (削除) しようとしています。パッチが適用されたファイルをパッチ適用前のバージョンに復元するには、最初のインストール時に使用した CD から元のファイルを復元してください。

メッセージ

Patch *patch_id* has not been successfully applied to this system.
Will remove directory *dir*.

説明・対処方法

システムに適用されていないパッチをバックアウト (削除) しようとしています。パッチは適用されていませんが、(失敗した `patchadd` によって作成された) `/var/sadm/patch/patch_id` ディレクトリが残っています。パッチをバックアウトすることはできません。パッチが適用されたファイルをパッチ適用前のバージョンに復元するには、最初のインストール時に使用した CD から元のファイルを復元してください。

メッセージ

This patch was obsoleted by patch *patch_id*.
Patches must be backed out in the reverse order in which they were installed. Patch backout aborted.

説明・対処方法

不適切な順番でパッチをバックアウト (削除) しようとしています。パッチは正しい順序でバックアウトする必要があります。バックアウトしようとしているパッチ以外のパッチにも影響が及んでいる可能性があります。

メッセージ

Patch *patch_id* is required to be installed by an already installed *patch_id*.
It cannot be backed out until the required patch is backed out first.

説明・対処方法

インストール (適用) されている必要があるとメッセージ中に示されているパッチをバックアウト (削除) してから、目的のパッチをバックアウトしてください。

patchrm(1M)

メッセージ

```
The installation of patch patch_id was interrupted.
```

説明・対処方法

以前に行なったパッチのインストールが中断されています。目的のパッチをバックアウト (削除) する前に、インストールが中断されたパッチをインストールする必要があります。

メッセージ

```
Patch patch_id was installed without backing up the original files.  
It cannot be backed out.
```

説明・対処方法

パッチの適用時に `patchadd` コマンドの `-d` オプションを指定したか、またはディスク領域を確保するためにパッチの保存領域が削除されています。このため、元のファイルが保存されていないので `patchrm` を使用できません。元のファイルはインストール CD からのみ復元できます。

メッセージ

```
pkgadd of pkgname package failed return code code.  
See /var/sadm/patch/patch_id/log for reason for failure.
```

説明・対処方法

メッセージに示されているパッチパッケージのインストールに失敗しています。ログファイルを参照して原因を確認してください。問題を修正後に、バックアウトスクリプトを再度実行してください。

メッセージ

```
Restore of old files failed.
```

説明・対処方法

パッチが適用されたファイルをパッチ適用前のバージョンに復元するために、バックアウトスクリプトが `cpio` コマンドを使用しています。上記のメッセージの前に `cpio` コマンドからの出力が表示されているはずですが、`cpio` が失敗した原因を調べて必要な処置を行なってください。これは、Solaris 2.4 およびそれ以前のバージョンのシステムで発生するエラーです。

関連項目 `cpio(1)`, `pkginfo(1)`, `patchadd(1M)`, `pkgadd(1M)`, `pkgchk(1M)`, `pkgrm(1M)`, `showrev(1M)`, `attributes(5)`

注意事項 クライアント・サーバーマシンでは、既存のクライアントからまたはクライアントのルートディレクトリ (`templates` 領域) からパッチパッケージは削除 (バックアウト) されません。このため、すべてのクライアントマシンにおいて直接 `patchrm` を使用して、クライアントマシンから適切な時にパッチを削除する必要があります。パッケージユーティリティ (`pkgadd`, `pkgrm`, `pkgchk`) に影響するバグは、`patchadd` または `patchrm` の動作に影響する場合があります。`patchadd` および `patchrm` は、上記のパッケージユーティリティを使用してパッチパッケージをインストールおよびバックアウトしています。これらのパッケージユーティリティのバグを修正する

patchrm(1M)

パッチが提供されているかどうかを確認し、提供されている場合はそのパッチを先に適用してから、他のパッチを適用することをお勧めします。現在提供されている、パッケージユーティリティに対するパッチは、次のとおりです。

Solaris 2.1:

100901

Solaris 2.2:

101122

Solaris 2.3:

101331

Solaris 2.4 (SPARC 版):

102039

Solaris 2.4 (Intel 版):

102041

Solaris 2.5.1 (SPARC 版):

104578

Solaris 2.5.1 (Intel 版):

104579

Solaris 2.6 (SPARC 版):

106292

Solaris 2.6 (Intel 版):

106293

pgxconfig(1M)

名前	pgxconfig, GFXconfig – PGX32 (Raptor GFX) グラフィックスアクセラレータの設定
形式	<pre>/usr/sbin/pgxconfig [-dev <i>device-filename</i>] [-res <i>video-mode</i> [try noconfirm nocheck]] [-file machine system] [-depth 8 24] [-24only] [-defaults] /usr/sbin/pgxconfig [-propt] [-prconf] /usr/sbin/pgxconfig [-help] [-res ?] /usr/sbin/pgxconfig [-i]</pre>
機能説明	<p>pgxconfig ユーティリティは PGX32 (Raptor GFX) グラフィックスアクセラレータの設定用コマンドで、これには X11 ウィンドウシステムの一部を PGX32 (Raptor GFX) 用にデフォルト設定する機能も含まれます。以前のバージョンでは、このユーティリティは GFXconfig という名称でした。</p> <p>形式の項に記された pgxconfig の第 1 の形式では、指定したオプションを OWconfig ファイルに保存します。次回の PGX32 (Raptor GFX) デバイス上でのウィンドウシステム起動時に、ここで指定したオプションに従って PGX32 (Raptor GFX) デバイスを初期化します。OWconfig ファイルに保存されたオプションの更新内容は、異なる複数のウィンドウシステムセッションや再起動後のシステムでも有効となります。</p> <p>その他の形式 (2、3、4 番目の形式) を使用した場合はオプション -prconf, -propt, -help, -res ? を呼び出すだけで、OWconfig ファイルに保存されているオプションを更新することはありません。更に、3 番目の形式を使用した場合はこれ以外のオプションはすべて無視されます。</p> <p>-i オプションを使用すると、pgxconfig は、対話型モードで起動します。</p> <p>一度にオプションを指定できる PGX32 (Raptor GFX) デバイスの数は、1 つだけです。複数の PGX32 (Raptor GFX) デバイ스에 オプションを指定する場合は、pgxconfig -i を複数回起動する必要があります。</p> <p>pgxconfig コマンドで指定できるのは、PGX32 (Raptor GFX) 固有のオプションだけです。デフォルトの表示色数、デフォルトのビジュアルクラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください。SUNWxwman パッケージに含まれている Xsun(1) マニュアルページを参照してください。</p> <p>ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは /usr/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/etc/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。</p> <p>いずれの標準 OWconfig ファイルでも、書き込み権限があるのはスーパーユーザーだけです。したがって、pgxconfig は、スーパーユーザーがプログラムの所有者となり、setuid による root 権限で実行されます。</p>

オプション

`-dev device-filename`

PGX32 (Raptor GFX) 特殊ファイルを指定します。デフォルトは `/dev/fbs/gfxp0`、または使用可能な場合であれば `/dev/fbs/raptor0` です。

`-file machine | system`

更新する OWconfig ファイルを指定します。ここで `machine` が指定された場合は、`/etc/openwin` ディレクトリツリーにあるマシン固有の OWconfig ファイルが使用されます。`system` が指定された場合は、`/usr/openwin` ディレクトリツリーにある共通の OWconfig ファイルが使用されます。ファイルがない場合は、新たに生成されます。

`-res video-mode [try | noconfirm | nocheck]`

PGX32 (Raptor GFX) デバイ스에接続されているモニターの制御に使用する組み込みのビデオモードを指定します。

ビデオモードの形式は、以下のいずれか 1 つになります。

`widthxheightxrate`

`width` はピクセル単位の画面の幅、`height` はピクセル単位の画面の高さ、`rate` は垂直方向の画面再描画周期です。`-res` では、再描画周期 `rate` の前の `x` は、`@` でも代用できます。つまり周期の指定に限っては、たとえば `1280x1024@76` のような形式にも対応しています。オプションに `-res ?` を付けて `pgxconfig` を実行すると (コマンド形式の項に記された 3 番目の形式)、ビデオモードの一覧が表示されます。ビデオボードとモニターの両方が、すべての解像度をサポートしているわけではありません。`noconfirm` または `nocheck` オプションを指定しないで、ボードのサポートしていない解像度を入力した場合には、`pgxconfig` はその要求を許可しません。また、`nocheck` オプションを指定しないで、モニターのサポートしていない解像度を指定した場合には、その値を適用する前に確認を求めてきます。

記号名 (Symbolic names)

便宜上、一部のビデオモードには記号名が定義されています。`widthxheightxrate` の形式の代わりに、記号名を `-res` の引数として指定することができます。記号名 `none` は、ウィンドウシステムを実行すると、画面の解像度が現在デバイスにプログラムされているビデオモードになることを意味します。

<code>svga</code>	<code>1024x768x60</code>
<code>1152</code>	<code>1152x900x76</code>
<code>1280</code>	<code>1280x1024x76</code>
<code>vga</code>	<code>640x480x60</code>
<code>none</code>	デフォルトのコンソールの解像度

pgxconfig(1M)

`-res` オプションには、ビデオモードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定できます。

`noconfirm` `-res` オプションを指定した際に、システムが使用可能であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために `pgxconfig` のデフォルトの動作では、この問題についての警告メッセージと、処理を継続するかどうかを確認するメッセージを表示します。`noconfirm` オプションを指定すると、`pgxconfig` コマンドはこの確認をせずに、要求のあったビデオモードにプログラムします。このオプションは、`pgxconfig` がシェルスクリプトから実行されている場合に便利です。

`nocheck` このオプションを指定すると、モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定されたビデオモードは、現在接続されているモニターに適切かどうかにかかわらず受け付けられます。このオプションは、PGX32 (Raptor GFX) デバイスに異なるモニターを接続する場合に便利です。このオプションの指定は、`noconfirm` の指定も兼ねます。

`try` このオプションを指定すると、指定したビデオモードを適用する前にテストすることができます。まず、指定したモードに基づいたテストパターンが表示されます。テストパターンが正常に表示された場合は、「y」(次いでキャリッジリターン)を入力します。「y」以外の文字を入力(次いでキャリッジリターンを入力)した場合は、「no」となります。

`-res ?`
PGX32 およびモニターがサポートする解像度の一覧を表示します。

`-24only`
Openwindows の実行時に、PGX32 (Raptor GFX) デバイスが 24 ビットカラーのみを使用するようにします。

`-defaults`
すべてのオプションの値をそれぞれのデフォルト値に戻します。

`-propt`
`-file` オプションで指定された `OWconfig` ファイルに書かれた PGX32 (Raptor GFX) オプションの値のうち、`-dev` オプションで指定されたデバイスに対するものすべてを表示します。`pgxconfig` の呼び出しが終了した後に、`OWconfig` ファイルに書き込まれるオプションの値を表示します。以下に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/gfxp0 ---
OWconfig: machine
Video Mode: not set
```

-prconf

PGX32 (Raptor GFX) のハードウェア構成を表示します。以下に表示例を示します。

```

--- Hardware Configuration for /dev/fbs/gfxp0 ---
DAC: version 0x0
Type:
Board:
PROM: version 0x0
PROM Information:
RAM:
EDID Data:
Monitor Sense ID:
Card possible resolutions:  640x480x60, 800x600x75, 1024x768x60
                            1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76
                            1280x1024x60, 1152x900x66, 1152x900x76, 1280x1024x67
                            960x680x112S, 960x680x108S, 640x480x60i, 768x575x50i,
                            1280x800x76, 1440x900x76, 1600x1000x66, 1600x1000x76,
                            vga, svga, 1152, 1280, stereo, ntsc, pal
Monitor possible resolutions: 720x400x70, 720x400x88, 640x480x60
                              640x480x67, 640x480x72, 640x480x75, 800x600x56,
                              800x600x60, 800x600x72, 800x600x75, 832x624x75,
                              1024x768x87, 1024x768x60, 1024x768x70, 1024x768x75,
                              1280x1024x75, 1280x1024x76, 1152x900x66, 1152x900x76,
                              1280x1024x67, 960x680x112S, vga, svga, 1152, 1280
                              stereo
Current resolution setting: 1280x1024x76
Possible depths:
Current depth: 8

```

-help

pgxconfig コマンド行のオプションと機能の概要を一覧で表示します。

-i

pgxconfig コマンドを対話型モードで起動します。

デフォルト

pgxconfig コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。ただし、`-depth` と `-24only` については、その限りではありません。

ウィンドウシステムを実行する際に、pgxconfig による PGX32 (Raptor GFX) のオプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルト値は次のとおりです。

```

-dev          /dev/fbs/gfxp0
-file         system
-res         none

```

`-res` オプションのデフォルト値 `none` とは、ウィンドウシステムが実行された場合に、画面解像度がそのデバイスに現在プログラムされているビデオモードになることを意味しています。

pgxconfig(1M)

使用例	<p>例 1 モニターの種類を変更する。</p> <p>モニターの種類を、垂直周波数 76 Hz で解像度 1280 x 1024 に変更する例を以下に示します。</p> <pre>example% /usr/sbin/pgxconfig -res 1280x1024x76</pre>
ファイル	<pre>/dev/fbs/gfxp0</pre> <p>デバイス特殊ファイル</p> <pre>/usr/openwin/server/etc/OWconfig</pre> <p>システム構成ファイル</p> <pre>/etc/openwin/server/etc/OWconfig</pre> <p>マシン構成ファイル</p>
関連項目	<p>PGX32 PCI グラフィックスカード インストールマニュアル</p>

名前	pmconfig – 電源管理システムの設定
形式	<code>/usr/sbin/pmconfig [-r] [-f file]</code>
機能説明	<p>pmconfig ユーティリティは、電源管理システムと保存停止・復元再開機能を設定します。ユーザーは、<code>/etc/default/power</code> のキーワードである <code>PMCHANGEPERM</code> に従って、許可されている場合にのみ pmconfig を使用して電源管理設定を変更する権限を持ちます。<code>/etc/default/power</code> のキーワードである <code>CPRCHANGEPERM</code> に従って、許可されている場合にのみ pmconfig を使用して保存停止・復元再開機能を変更する権限を持ちます。<code>/etc/default/power</code> のキーワードである <code>PMCHANGEPERM</code> と <code>CPRCHANGEPERM</code> の詳細については以下に記述する「ファイル」セクションを参照してください。</p> <p>ユーザーの権限に基づいて、pmconfig は、最初に電源管理システムまたは保存停止・復元再開 (またはその両方) をリセットします。それから新しい電源管理システムまたは保存停止・復元再開 (またはその両方) の設定を <code>/etc/power.conf</code> から読み取り、この新しい設定を有効にするコマンドを発行します。pmconfig ユーティリティはシステムの起動時に実行されます。また、このユーティリティは、<code>/etc/power.conf</code> ファイルに対する手動による変更の後でコマンド行から実行することもできます。<code>/etc/power.conf</code> ファイルを編集した場合、その変更を有効にするには、pmconfig を実行する必要があります。</p>
オプション	<p>以下のオプションを使用することができます。</p> <p><code>-r</code> 電源管理システムと保存停止・復元再開状態をデフォルトの状態にリセットして終了します。このオプションを使用するには、電源管理システムと保存停止・復元再開の設定に対する権限が必要です。</p> <p><code>-f file</code> ユーザーの権限に基づいて、pmconfig は、最初に電源管理システムまたは保存停止・復元再開 (またはその両方) をリセットします。それから新しい電源管理システムまたは保存停止・復元再開 (またはその両方) の設定を <code>/etc/power.conf</code> の代わりに <code>file</code> から読み取り、この新しい設定を有効にするためにコマンドを発行します。pmconfig が電源管理および (または) 保存停止・復元再開機能の設定ができた場合、<code>/etc/power.conf</code> 内の対応する設定は、<code>file</code> 内の設定に置き換えられます。</p>
終了ステータス	<p>以下の終了ステータスが返されます。</p> <p>0 正常終了</p> <p>>0 エラーが発生</p>
ファイル	<p><code>/etc/power.conf</code> システムの電源管理設定ファイル</p> <p><code>/etc/default/power</code> システムの電源管理システムと保存停止・復元再開機能に対する権限を制御するファイル。<code>PMCHANGEPERM</code> キーワードは電源管理設定の権限を制御し、<code>CPRCHANGEPERM</code> キーワードは保存停止・復元再開機能の設定権限を制御します。</p> <p>以下に指定できる値を示します。</p>

pmconfig(1M)

all 全てのユーザーが設定の変更ができます。

- スーパーユーザーだけが設定の変更ができます。

<user1, user2, ...> このユーザーリストに指定されているユーザーまたはスーパーユーザーは、設定の変更ができます。このリストには、空白またはコンマ (,) を使用して複数のユーザーを指定できます。このリストは、< と > で囲む必要があります。

console-owner システムコンソールデバイスノードを所有するユーザーまたはスーパーユーザーだけが設定を変更できません。

デフォルトの値は、PMCHANGEPERM=console-owner
とCPRCHANGEPERM=console-owner です。

属性 以下の属性については、attributes(5) を参照してください。

属性タイプ	属性値
使用条件	SUNWpmu
インタフェースの安定性	不安定

関連項目 dtpower(1M), powerd(1M), power.conf(4), attributes(5), cpr(7), pm(7D)

診断 プログラムが設定ファイルを開くことができない場合は、標準エラー出力にエラーメッセージを出力します。プログラムが設定ファイル内の構文エラーを検出した場合は、エラーメッセージと設定ファイル内のエラーの行番号を出力します。プログラムは、その行の残りの情報は処理せずに次の行を処理します。エラーを含んでいる行の設定情報の中で、すでに処理されているものは使用されます。電源管理システムおよび(または)保存停止・復元再開機能の設定を変更する権限がない場合、また設定ファイル中にユーザーが権限を持っていないエントリがある場合、権限を持つエントリについてだけ処理が行われ、権限のない部分についてはエラーが出力されます。

名前	powerd – 電源管理デーモン						
形式	<code>/usr/lib/power/powerd [-n]</code>						
機能説明	<p>powerd デーモンは pmconfig(1M) で起動し、システム動作の監視、および保存停止 - 復元再開機能を使用した自動停止を行います。システムが保存停止されると、電源が落とされる前に、完全な現在の状態がディスクに保存されます。リブート時に、システムは自動的に復元再開操作を開始し、システムが保存停止する直前と同じ状態に復元されます。</p> <p>システムの停止の直前に、powerd デーモンは停止について syslogd(1M) に通知し、syslogd(1M) はこれをブロードキャストします。</p>						
オプション	<p>以下のオプションを使用することができます。</p> <p><code>-n</code> 通知なし。デーモンは syslogd(1M) に通知しないでシステムを停止します。</p>						
ファイル	<code>/etc/power.conf</code> 電源管理設定情報ファイル						
属性	以下の属性については、attributes(5) を参照してください。						
	<table border="1"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWpmu</td> </tr> <tr> <td>インタフェースの安定性</td> <td>不安定</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWpmu	インタフェースの安定性	不安定
属性タイプ	属性値						
使用条件	SUNWpmu						
インタフェースの安定性	不安定						
関連項目	<p>pmconfig(1M), syslogd(1M), power.conf(4), attributes(5), cpr(7), pm(7D)</p> <p>『電源管理システムユーザーマニュアル』</p>						

prodreg(1M)

名前 prodreg – Solaris Product Registry の GUI ビューア

形式 /usr/bin/prodreg [subcommand]

機能説明 prodreg は、Solaris Product Registry (ProdReg) のビューアです。Solaris Product Registry は、Solaris オペレーティング環境のシステム上にインストールしたソフトウェア製品を記録して管理するためのシステムです。

オペラント 次のオペラントを指定できます。

subcommand prodreg のサブコマンドを指定します。サブコマンドを省略した場合は、swing サブコマンドを指定したのと同義になります。

使用法 次のサブコマンドを指定できます。

swing Java 2 SDK, Standard Edition の Swing グラフィカルユーザーインタフェース (デフォルト) を使用して、ビューアを起動します。prodreg コマンドに何もサブコマンドを指定しない場合と同義です。

awt Java AWT グラフィカルユーザーインタフェースを使用して、ビューアを起動します。

help 使用方法についてのメッセージを出力します。

version 現在の prodreg プログラムのバージョンを出力します。

環境 次の環境変数は、prodreg の実行に影響を与えます。

PKG_INSTALL_ROOT システムの PKG_INSTALL_ROOT パスとして使用するディレクトリの完全パス名を定義します。この環境変数が定義されている場合、すべての製品およびパッケージの情報ファイルは、最初に PKG_INSTALL_ROOT パスにおいて検索されます。この環境変数が定義されていない場合、デフォルトのシステムパス / が使用されます。

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWwsrv

関連項目 installer(1M), attributes(5)

名前	prtdiag – システムの診断情報の出力				
形式	<code>/usr/platform/platform-name/sbin/prtdiag [-v] [-1]</code>				
機能説明	<p>prtdiag は、sun4u システムおよび sun4d システム上で、システム設定と診断情報を表示します。</p> <p>診断情報は、システム内で障害の発生した現場交換可能ユニット(FRU) を表示します。</p> <p>prtdiag に必要なインタフェース、出力、およびディレクトリ階層における位置は、まだ確定されていません。将来のリリースで変更される可能性があります。</p> <p><i>platform-name</i> は、実装するプラットフォームの名前です。uname(1) に <code>-i</code> オプションを付けて実行すると表示されます。</p> <p>注:prtdiag は、Sun Enterprise 10000 サーバーで実行された場合は、診断情報と環境状態を表示しません。それらの情報を得るには、システムサービスプロセッサ(SSP)上の <code>/var/opt/SUNWssp/adm/\${SUNW_HOSTNAME}/messages</code> ファイルを参照してください。</p>				
オプション	<p>以下のオプションを使用することができます</p> <p><code>-v</code> 詳細表示モード。最近発生した AC 電源障害の時刻、最近発生した重大なハードウェアエラーの情報、および (必要に応じて) 環境状態を表示します。重大なハードウェアエラーの情報は、FRU を修理する場合や、詳細な診断を行う場合に有用です。</p> <p><code>-1</code> ログの出力。システムで障害またはエラーが発生した場合は、syslogd(1M) のみに対してこの情報を出力します。</p>				
終了ステータス	<p>以下の終了値が返されます。</p> <p>0 システム内で何も異常が検出されなかった。</p> <p>1 システム内で異常が検出された。</p> <p>2 メモリー不足などの、内的な prtdiag のエラーが発生した。</p>				
属性	以下の属性については、attributes(5) を参照してください。				
	<table border="1"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWkvm</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWkvm
属性タイプ	属性値				
使用条件	SUNWkvm				
関連項目	uname(1), modinfo(1M), prtconf(1M), psrinfo(1M), sysdef(1M), syslogd(1M), attributes(5), openprom(7D)				

raidctl(1M)

名前	raidctl - RAID ハードウェアユーティリティー
形式	raidctl -c disk1 disk2 raidctl -d disk1 raidctl [-f] -F filename controller... raidctl -l [controller...]
機能説明	<p>raidctl ユーティリティーは、LSI1030 HW RAID コントローラの RAID ボリュームを作成、削除、または表示します。</p> <p>raidctl ユーティリティーには、基本ファイルシステムのアクセス権で制御される特権が必要です。特権を持つユーザーだけが RAID システム構成を操作できます。特権のないユーザーが RAID ボリュームを作成または削除しようとする、コマンドは EPERM で失敗します。</p> <p>オプションを指定しないと、raidctl は、すべての既存のコントローラに関して現在の RAID 構成を表示します。</p>
オプション	<p>次のオプションがサポートされています。</p> <p>-c disk1 disk2 <i>disk1</i> と <i>disk2</i> を使用してミラーを作成します。<i>disk2</i> の内容を <i>disk1</i> の内容で置換します。<i>disk1</i> と <i>disk2</i> を標準的な形式 (たとえば、c0t0d0) で指定します。</p> <p>RAID ボリュームを作成するとき、RAID ボリュームはディスクペアの最初のターゲット (<i>disk1</i>) の識別情報を前提とします。2 番目のターゲット (<i>disk2</i>) はシステムから消失します。このため、RAID ボリュームは1つのディスクとして表示されます。</p> <p>RAID の作成時に、指定されたコントローラに RAID 構成がすでに存在している場合は失敗します。さらに、2 番目のディスクは、すべてのデータが消去されて、1 番目のディスクのデータで置換されるので、マウントしないでください。</p> <p>-d disk1 <i>disk1</i> として指定された RAID ボリュームを削除します。</p> <p><i>disk1</i> を標準的な形式 (たとえば、c0t0d0) で指定します。</p> <p>-f 強制的に更新します。プロンプトは出力しません。</p> <p>-F filename controller 指定のコントローラ (<i>controller</i>) で動作するファームウェアを更新します。</p> <p>-l [controller ...] システムの RAID 構成を一覧表示します。<i>controller</i> を指定した場合は、<i>controller</i> の RAID 構成が一覧表示されます。</p>

-l による出力は、次の情報を一覧表示します。

RAID Volume	論理 RAID ボリューム名を表示します。
RAID Status	RESYNCING (ディスクが同期している)、DEGRADED (RAID は機能を制限して稼動している)、OK (正常に稼動している)、または FAILED (機能していない) のいずれかの RAID ステータスを表示します。
RAID Disk	RAID ディスク名を表示します。
Disk Status	OK または FAILED のいずれかのディスクステータスを表示します。

使用例 例 1 RAID 構成の作成

次のコマンドにより RAID 構成を作成します。

```
# raidctl -c c0t0d0 c0t1d0
RAID Volume 'c0t0d0' created
```

例 2 RAID 構成の表示

次のコマンドにより RAID 構成を表示します。

```
# raidctl
RAID      RAID      RAID      Disk
Volume    Status    Disk      Status
-----
c0t0d0    RESYNCING c0t0d0    OK
                c0t1d0    OK
```

例 3 RAID 構成の削除

次のコマンドにより、RAID 構成を削除します。

```
# raidctl -d c0t0d0
RAID Volume 'c0t0d0' deleted.
```

例 4 コントローラのフラッシュイメージの更新

次のコマンドにより、コントローラのフラッシュイメージを更新します。

```
# raidctl -F lsi1030.fw 0
Update flash image on controller 0? (y/N): y
Flash updated successfully
```

終了ステータス 次の終了値が返されます。

raidctl(1M)

- 0 正常に終了しました。
- 1 不正なコマンド行入力です。
- 2 要求操作は失敗しました。

属性 以下の属性については、attributes(5)を参照してください。

属性タイプ	属性値
Availability	SUNWcsu

関連項目 attributes(5)
Solaris のシステム管理 (第 1 巻)

名前	accept, reject – 印刷要求の受付または拒否						
形式	accept <i>destination</i> ... reject [-r <i>reason</i>] <i>destination</i> ...						
機能説明	accept コマンドは、指定した宛先への印刷要求を待ち行列に加えることを許可します。 reject コマンドは、指定した宛先への印刷要求を待ち行列に加えることができないようにします。 宛先が印刷要求を受け付けたか、または拒否したかを確認するには、lpstat -a を実行してください。 accept と request は印刷サーバー上で実行する必要があります。クライアントシステム上では意味をもちません。						
オプション	reject コマンドには、次のオプションを指定できます。 -r <i>reason</i> <i>destination</i> への印刷要求を抑止する理由を文字列として記述します。 <i>reason</i> に空白が含まれる場合は引用符で囲ってください。 <i>reason</i> は、プリンタの状況を調べる lpstat -a コマンドの出力中に表示されます。 <i>reason</i> のデフォルト値は、既存の宛先に関しては unknown reason、システムに追加されたばかりでまだ印刷要求を受け付けたことのない宛先に関しては new printer となります。						
オペランド	次のオペランドを指定できます。 <i>destination</i> 印刷要求を受け付ける、または拒否する宛先名。宛先にはプリンタ名やプリンタクラスを指定します (lpadmin(1M) 参照)。 <i>destination</i> は名前を使用して指定します。名前の命名規約については printers.conf(4) のマニュアルページを参照してください。						
終了ステータス	以下の終了ステータスが返されます。 0 正常終了 0 以外 エラーが発生した						
ファイル	/var/spool/lp/* LP 印刷待ち行列						
属性	次の属性については attributes(5) のマニュアルページを参照してください。						
	<table border="1"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWpcu</td> </tr> <tr> <td>CSI</td> <td>対応済み (「注意事項」参照)</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWpcu	CSI	対応済み (「注意事項」参照)
属性タイプ	属性値						
使用条件	SUNWpcu						
CSI	対応済み (「注意事項」参照)						

reject(1M)

関連項目	enable(1), lp(1), lpstat(1), lpadmin(1M), lpsched(1M), printers.conf(4), attributes(5)
注意事項	<p>accept と reject は印刷サーバーのスプール用システムの待ち行列に対してのみ有効です。したがって、クライアントシステムから出された要求については、印刷サーバーのスプール用システムが取り消したり受け取ったりするまでは、クライアントシステムの印刷待ち行列に入れられたままの状態になります。</p> <p>accept は <i>destinations</i> 名を除いて CSI 対応が可能です。</p>

名前	rsh, restricted_shell – 制限付きシェルのコマンドインタプリタ
形式	<code>/usr/lib/rsh [-acefhiknprstuvx] [argument...]</code>
機能説明	<p>rsh は、標準のコマンドインタプリタである sh と比べて機能の一部が制限されており、ログインが許される実行環境も、sh の環境と比べて制限されています。機能の詳細や使用方法に関しては、sh(1) の説明を参照してください。</p> <p>シェルは、呼び出されると環境変数 SHELL を調べます。この環境変数が存在していて、その値のファイル名部分が rsh であれば、シェルは制限付きシェルとなります。</p> <p>rsh の機能は、以下の動作ができない点を除き sh と同一です。</p> <ul style="list-style-type: none"> ■ ディレクトリの変更 (cd(1) を参照) ■ \$PATH の値の設定 ■ / を含むパスまたはコマンド名の指定 ■ 出力先のリダイレクト (> および >>) <p>これらの制限は、.profile の解釈後に有効となります。</p> <p>制限付きシェルは、次のいずれかの方法で呼び出せます。</p> <ol style="list-style-type: none"> (1) /etc/passwd ファイルの最後のエントリのファイル名部分を rsh と記述する (passwd(4) を参照) (2) 環境変数 SHELL が存在していて、その値のファイル名の部分が rsh である。環境変数 SHELL は、.login ファイル内に設定されている必要がある (3) シェルの呼び出し時に、引数 0 のファイル名の部分が rsh である (4) シェルを -r オプション付きで呼び出す <p>実行するコマンドがシェル手続きである場合は、rsh は sh を呼び出して、コマンドを実行します。したがって、利用できるコマンドの種類には制限はありますが、一般ユーザーは標準シェルの全機能を利用できるシェル手続きを使用することができません。このスキーマは、一般ユーザーが同じディレクトリにおいて書き込み権と実行権を持っていないことを想定しています。</p> <p>.profile の作成者 (profile(4) を参照) が、確実な設定処理を実行してユーザーを適切なディレクトリ (おそらく、ログインディレクトリではない) に置くことにより、ユーザーの動作を完全に制御できるという点が、これらの規約の実際の効果となります。</p> <p>システム管理者は、制限付きシェルで安全に起動できるコマンドのディレクトリ (つまり /usr/rbin) を設定することがよくあります。システムによっては、制限付きエディタ red を提供するものもあります。</p>
終了ステータス	<p>構文エラーなどのエラーを検出した場合、シェルは 0 以外の終了ステータスを返します。シェルを対話型以外で使用している場合、シェルフファイルの実行は中止されます。対話型で使用している場合は、シェルは最後に実行されたコマンドの終了ステータスを返します。</p>

restricted_shell(1M)

属性 次の属性については `attributes(5)` のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWcsu

関連項目 `cd(1)`, `intro(1)`, `login(1)`, `rsh(1)`, `sh(1)`, `exec(2)`, `passwd(4)`, `profile(4)`, `attributes(5)`

注意事項 制限付きシェル `/usr/lib/rsh` を、リモートシェル `/usr/bin/rsh` と混同しないように注意してください。リモートシェルに関しては `rsh(1)` を参照してください。

名前	rsh, restricted_shell – 制限付きシェルのコマンドインタプリタ
形式	<code>/usr/lib/rsh [-acefhiknprstuvx] [argument...]</code>
機能説明	<p>rsh は、標準のコマンドインタプリタである sh と比べて機能の一部が制限されており、ログインが許される実行環境も、sh の環境と比べて制限されています。機能の詳細や使用方法に関しては、sh(1) の説明を参照してください。</p> <p>シェルは、呼び出されると環境変数 SHELL を調べます。この環境変数が存在していて、その値のファイル名部分が rsh であれば、シェルは制限付きシェルとなります。</p> <p>rsh の機能は、以下の動作ができない点を除き sh と同一です。</p> <ul style="list-style-type: none"> ■ ディレクトリの変更 (cd(1) を参照) ■ \$PATH の値の設定 ■ / を含むパスまたはコマンド名の指定 ■ 出力先のリダイレクト (> および >>) <p>これらの制限は、.profile の解釈後に有効となります。</p> <p>制限付きシェルは、次のいずれかの方法で呼び出せます。</p> <ol style="list-style-type: none"> (1) /etc/passwd ファイルの最後のエントリのファイル名部分を rsh と記述する (passwd(4) を参照) (2) 環境変数 SHELL が存在していて、その値のファイル名の部分が rsh である。環境変数 SHELL は、.login ファイル内に設定されている必要がある (3) シェルの呼び出し時に、引数 0 のファイル名の部分が rsh である (4) シェルを -r オプション付きで呼び出す <p>実行するコマンドがシェル手続きである場合は、rsh は sh を呼び出して、コマンドを実行します。したがって、利用できるコマンドの種類には制限はありますが、一般ユーザーは標準シェルの全機能を利用できるシェル手続きを使用することができません。このスキーマは、一般ユーザーが同じディレクトリにおいて書き込み権と実行権を持っていないことを想定しています。</p> <p>.profile の作成者 (profile(4) を参照) が、確実な設定処理を実行してユーザーを適切なディレクトリ (おそらく、ログインディレクトリではない) に置くことにより、ユーザーの動作を完全に制御できるという点が、これらの規約の実際の効果となります。</p> <p>システム管理者は、制限付きシェルで安全に起動できるコマンドのディレクトリ (つまり /usr/rbin) を設定することがよくあります。システムによっては、制限付きエディタ red を提供するものもあります。</p>
終了ステータス	<p>構文エラーなどのエラーを検出した場合、シェルは 0 以外の終了ステータスを返します。シェルを対話型以外で使用している場合、シェルフファイルの実行は中止されます。対話型で使用している場合は、シェルは最後に実行されたコマンドの終了ステータスを返します。</p>

rsh(1M)

属性 次の属性については `attributes(5)` のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWcsu

関連項目 `cd(1)`, `intro(1)`, `login(1)`, `rsh(1)`, `sh(1)`, `exec(2)`, `passwd(4)`, `profile(4)`, `attributes(5)`

注意事項 制限付きシェル `/usr/lib/rsh` を、リモートシェル `/usr/bin/rsh` と混同しないように注意してください。リモートシェルに関しては `rsh(1)` を参照してください。

名前	scadm – システムコントローラ (SC) の管理
形式	scadm <i>subcommand</i> [<i>option</i>] [<i>argument...</i>]
機能説明	<p>scadm ユーティリティは、システムコントローラ (SC) を管理します。このユーティリティを使用すると、ホストサーバーが SC と対話できるようになります。</p> <p>scadm ユーティリティは、スーパーユーザーとして実行しなければなりません。</p> <p>scadm ユーティリティには 15 個のサブコマンドがあります。一部のサブコマンドには、そのサブコマンドに関連付けられている特定のオプションと引数があります。「サブコマンド」、「オプション」および「オペランド」を参照してください。</p>
サブコマンド	<p>コマンド行の scadm コマンドの後に SPACE を入力し、その後にサブコマンドを指定します。</p> <p>次のサブコマンドを指定できます。</p> <p>date SC の日付と時刻を表示します。</p> <p>date サブコマンドの書式を以下に示します。</p> <pre>scadm date</pre> <p>download SC のファームウェアをプログラムします。</p> <p>ファームウェアについては、ブートモニターとメインイメージの 2 つの部分があります。</p> <p>デフォルトでは、scadm コマンドのダウンロードを実行すれば、メインファームウェアイメージがプログラムされます。boot 引数を指定すると、ブートモニターのプログラミングが選択されません。</p> <p>download サブコマンドの書式を以下に示します。</p> <pre>scadm download [boot] file</pre> <p>help コマンドの一覧を表示します。</p> <p>help サブコマンドの書式を以下に示します。</p> <pre>scadm help</pre> <p>loghistory SC イベントログの最新のエンタリを表示します。</p> <p>loghistory サブコマンドの書式を以下に示します。</p> <pre>scadm loghistory</pre> <p>resetrsc SC をリセットします。実行できるリセットには、ハードリセットとソフトリセットの 2 種類があります。デフォルトで実行されるのがハードリセットです。ソフトリセットを選択する場合は -s オプションを指定します。</p>

scadm(1M)

	<p>resetrsc サブコマンドの書式を以下に示します。</p> <pre>scadm resetrsc [-s]</pre>
send_event	<p>テキストベースのイベントを手動で送信します。SC は、イベントを SC イベントログに転送できます。-c オプションを構成して、重要な警告を電子メール、ログイン中の SC ユーザー、および syslog に送信できます。重要なイベントは syslog(3C) に記録されます。関連テキストメッセージの文字数は 80 文字以内です。</p> <p>send_event サブコマンドの書式を以下に示します。</p> <pre>scadm send_event [-c] "message"</pre>
set	<p>SC 構成変数の値を設定します。</p> <p>SC 構成変数には、SC IP アドレス netsc_ipaddr や、SC カスタマ情報 sc_customerinfo などがあります。SC 構成変数の全リストについては、scadm help コマンドの出力結果を参照してください。</p> <p>set サブコマンドの書式を以下に示します。</p> <pre>scadm set variable value</pre>
show	<p>SC 構成変数の現在の設定を表示します。変数を指定しないで scadm を実行すると、すべての変数設定が表示されます。</p> <p>show サブコマンドの書式を以下に示します。</p> <pre>scadm show [variable]</pre>
shownetwork	<p>SC の現在のネットワーク構成パラメタを表示します。</p> <p>shownetwork サブコマンドの書式を以下に示します。</p> <pre>scadm shownetwork</pre>
useradd	<p>SC にユーザーアカウントを追加します。SC では 16 名までの個別ユーザーがサポートされます。</p> <p>useradd サブコマンドの書式を以下に示します。</p> <pre>scadm useradd username</pre>
userdel	<p>SC からユーザーアカウントを削除します。</p> <p>userdel サブコマンドの書式を以下に示します。</p> <pre>scadm userdel username</pre>
userpassword	<p>指定したユーザーアカウントのパスワードを設定します。このパスワードは、現在設定されている既存のパスワードがあった場合にはそれより優先されます。新しいパスワードを設定する前に、古いパスワードは検証されません。</p>

	<p>userpassword サブコマンドの書式を以下に示します。</p> <pre>scadm userpassword <i>username</i></pre>
userperm	<p>ユーザーのアクセス権のレベルを設定します。</p> <p>userperm サブコマンドの書式を以下に示します。</p> <pre>scadm userperm <i>username</i> [<i>aucr</i>]</pre>
usershow	<p>指定したユーザーアカウントの詳細を表示します。ユーザー名を指定しないと、すべてのユーザーアカウントの詳細が表示されません。</p> <p>usershow サブコマンドの書式を以下に示します。</p> <pre>scadm usershow <i>username</i></pre>
version	<p>SC のバージョン番号とそのコンポーネントを表示します。</p> <p>version サブコマンドの書式を以下に示します。</p> <pre>scadm version [-v]</pre>
オプション	<p>resetrsc、send_event、および version サブコマンドには、関連するオプションがあります。コマンド行のサブコマンドの後に SPACE を入力し、その後にオプションを指定します。</p> <p>resetrsc サブコマンドでは次のオプションを指定できます。</p> <p>-s ハードリセットの代わりにソフトリセットを実行します。ハードリセットは SC ハードウェアを物理的にリセットします。ソフトリセットの場合は、SC ソフトウェアがブートファームウェアにジャンプして、リセットをシミュレーションします。</p> <p>send_event サブコマンドでは次のオプションを指定できます。</p> <p>-c 重要なイベントを送信します。-c を指定しないで -send_event を実行すると、警告が送信されます。</p> <p>version サブコマンドでは次のオプションを指定できます。</p> <p>-v バージョン番号と関連情報の詳細な内容が表示されます。</p>
オペランド	<p>download, send_event, set, show, useradd, userdel, userperm, usershow, userpassword, userperm の各サブコマンドには関連する引数 (オペランド) があります。</p> <p>サブコマンドにオプションを指定する場合は、コマンド行のオプションの後に SPACE を入力し、その後に引数を指定します。サブコマンドにオプションを指定しない場合は、コマンド行のサブコマンドの後に SPACE を入力し、その後に引数を指定します。引数を 2 つ以上指定する場合は、それぞれの引数の後に SPACE を入力して区切ります。</p>

scadm(1M)

download サブコマンドでは次の引数を指定できます。

boot フラッシュのブートモニター部分をプログラムします。フラッシュのメイン部分は、引数ファイルを使用せずにプログラムされます。

file ダウンロード用のブートまたはメインファームウェアイメージの格納場所へのパスを *file* で指定します。

file の例は、以下のとおりです。

```
/usr/platform/platform_type/lib/image/alommainfw
```

または

```
/usr/platform/platform_type/lib/image/alombootfw
```

send_event サブコマンドでは次の引数を指定できます。

"message" *message* に指定されているテキストを使用して、イベントを記述します。*message* は引用符で囲みます。

set サブコマンドでは次の引数を指定できます。

variable SC 構成 *variable* を設定します。

value SC 構成変数の *value* を設定します。

show サブコマンドでは次の引数を指定できます。

variable 指定した特定の変数の値を表示します。

useradd サブコマンドでは次の引数を指定できます。

username 新しい SC アカウント *username* を追加します。

userdel サブコマンドでは次の引数を指定できます。

username SC アカウント *username* を削除します。

userperm サブコマンドでは次の引数を指定できます。

aucr SC ユーザーアカウントのアクセス権を設定します。アクセス権を指定しない場合、4つのアクセス権はすべて無効になり、読み取り専用アクセス権が割り当てられます。

アクセス権の定義を以下に示します。

a ユーザーは SC 構成変数を管理または変更できません。

u ユーザーは、ユーザーコマンドを使用して SC アカウントを変更できます。

c ユーザーはコンソールに接続できます。

r ユーザーは SC のリセットと、ホストの電源投入および切断ができます。

username SC アカウント *username* に対するアクセス権を変更します。

usershow サブコマンドでは次の引数を指定できます。

username SC アカウント *username* に関する情報を表示します。*username* を指定しないと、すべてのアカウントの情報が表示されます。

userpassword サブコマンドでは次の引数を指定できます。

username *username* に対する SC パスワードを設定します。

userperm サブコマンドでは次の引数を指定できます。

username *username* に対する SC アクセス権を変更します。

使用例 例 1 SC の日付と時刻を表示する

次のコマンドは、SC の日付と時刻を表示します。

```
scadm date
```

例 2 SC の構成変数を設定する

次のコマンドは、SC の構成変数 `netsc_ipaddr` を `192.168.1.2` に設定します。

```
scadm set netsc_ipaddr 192.168.1.2
```

例 3 SC の現在の構成設定を表示する

次のコマンドは、SC の現在の構成設定を表示します。

```
scadm show
```

例 4 変数の現在の設定を表示する

次のコマンドは、`sys_hostname` という変数の現在の設定を表示します。

```
scadm show sys_hostname
```

例 5 テキストベースの重要なイベントを送信する

次のコマンドは、重要なイベントを SC イベントログに送信し、現在の SC ユーザーに警告するとともに、イベントを `syslog(3C)` に送信します。

```
scadm send_event -c "The UPS signaled a loss in power"
```

例 5 テキストベースの重要なイベントを送信する (続き)

例 6 参照用のテキストベースのイベントを送信する

次のコマンドは、重要ではない参照用のテキストベースのイベントを SC イベントログに送信します。

```
scadm send_event "The disk is close to full capacity"
```

例 7 ユーザーを SC に追加する

次のコマンドは、ユーザー `rscroot` を SC に追加します。

```
scadm useradd rscroot
```

例 8 SC からユーザーを削除する

次のコマンドは、ユーザー `olduser` を SC から削除します。

```
scadm userdel olduser
```

例 9 ユーザーの詳細を表示する

次のコマンドは、すべてのユーザーアカウントの詳細を表示します。

```
scadm usershow
```

例 10 特定のユーザーの詳細を表示する

次のコマンドは、ユーザーアカウント `rscroot` の詳細を表示します。

```
scadm usershow rscroot
```

例 11 ユーザーのアクセス権のレベルを設定する

次のコマンドは、ユーザー `rscroot` に全アクセス権 `aucr` を設定します。

```
scadm userperm rscroot aucr
```

例 12 ユーザーのアクセス権のレベルを設定する

次のコマンドは、ユーザー `newuser` にコンソールアクセス権 `c` だけを設定します。

```
scadm userperm newuser c
```

例 13 ユーザーのアクセス権のレベルを設定する

次のコマンドは、ユーザー `newuser` のアクセス権を読み取り専用を設定します。

```
scadm userperm newuser
```

例 13 ユーザーのアクセス権のレベルを設定する (続き)

例 14 現在のネットワークパラメタを表示する

次のコマンドは、SC の現在のネットワーク構成パラメタを表示します。

```
scadm shownetwork
```

例 15 ログ履歴を表示する

次のコマンドは、SC イベントログの最新のエントリを表示します。

```
scadm loghistory
```

例 16 詳細情報を表示する

次のコマンドは、SC とそのコンポーネントの詳細なバージョン情報を表示します。

```
scadm version -v
```

終了ステータス

次の終了値が返されます。

0 正常に終了。

0 以外 エラーが発生。

属性

以下の属性については、attributes(5) を参照してください。

属性タイプ	属性値
Availability	SUNWkvm

関連項目

syslog(3C)、attributes(5)

su(1M)

名前	su - スーパーユーザーまたは別のユーザーに変更
形式	su [-] [<i>username</i> [<i>arg...</i>]]
機能説明	<p>su コマンドを使用すると、ログオフをしないで別のユーザーまたは別の役割になることができます。デフォルトの <i>username</i> は root (スーパーユーザー) です。</p> <p>su を使用するときには、適切なパスワードを入力する必要があります (ユーザーがすでに root の場合は不要)。パスワードが正しければ、指定した <i>username</i> 用に設定されている実ユーザー ID、実効ユーザー ID、グループ ID、補助グループリストを持つ新しいシェルプロセスが生成されます。新しいシェルは、<i>username</i> のパスワードファイルエントリ (passwd(4) を参照) のシェルフールドで指定されたシェルです。シェルが指定されていないと、/usr/bin/sh (sh(1) を参照) になります。元のユーザー ID の権限に戻る場合は、EOF 文字 (CTRL-D) を入力して新しいシェルを終了します。</p> <p>コマンド行に引数を指定すると、その引数が新しいシェルに渡されます。sh などのプログラムを実行している場合は、<i>arg</i> (引数) に <i>-c string</i> と指定すると、シェルによって <i>string</i> が実行されます。また、引数に <i>-r</i> を指定すると、ユーザーに制限付きのシェルが与えら提供されます。</p> <p>次の説明は、指定したユーザーのパスワードファイルのエントリに、ログインシェルとして /usr/bin/sh が指定されている場合と、何も指定されていない場合にだけのみ有効です。後者の場合、ログインシェルはデフォルトで /usr/bin/sh になります。su の最初の引数に <i>-</i> (ダッシュ) を指定すると、指定したユーザーとして実際にログインした場合と同じ環境が渡されます。最初の引数に <i>-</i> (ダッシュ) を指定しない場合、\$PATH 以外の環境が渡されます。\$PATH は /etc/default/su 中で PATH と SUPATH によって制御されます。また、引数に <i>-</i> (ダッシュ) を指定した場合は、ユーザーのプロジェクト ID が設定されます。settaskid(2) のマニュアルページを参照してください。</p> <p>su を実行して別のユーザーになる操作は、すべてログファイル /var/adm/sulog に記録されます (sulog(4) を参照)。</p>
セキュリティ	<p>su は、pam(3PAM) を使って、認証、アカウント管理、セッション管理を行います。su で使用するモジュールを指定する PAM 構成ポリシーは、/etc/pam.conf に記述されています。次に、su コマンド (UNIX 認証、アカウント管理、セッション管理モジュールを実行) エントリが記述されている pam.conf ファイルの抜粋を示します。</p> <pre>su auth required /usr/lib/security/pam_unix.so.1 su account required /usr/lib/security/pam_unix.so.1 su session required /usr/lib/security/pam_unix.so.</pre> <p>su サービスのエントリがない場合は other のサービスのエントリを使用します。複数の認証モジュールが記述されている場合、複数のパスワードが必要になることがあります。</p>

使用例	<p>例 1 以前にエクスポートした環境を維持してユーザー bin になる</p> <p>以前にエクスポートした環境のままユーザー bin になるには、以下のコマンドを実行します。</p> <pre>example% su bin</pre> <p>例 2 ユーザー bin になり bin のログイン環境に変更する</p> <p>ユーザー bin になり、bin がログインした場合と同じ環境に切り替えるには、以下のコマンドを実行します。</p> <pre>example% su - bin</pre> <p>例 3 ユーザー bin の環境とアクセス権でコマンドを実行する</p> <p>ユーザー bin の一時的な環境とアクセス権を使ってコマンド <i>command</i> を実行する場合は、以下のように入力します。</p> <pre>example% su - bin -c "command args"</pre>
環境	<p>セキュリティ上の理由から、接頭辞 LD_ が付いている環境変数は削除されました。このため、以前にユーザー bin でエクスポートした接頭辞 LD_ 付きの環境変数は、su bin を実行しても使用できません。</p> <p>LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) が設定されていない環境では、環境変数 LANG によって、各ロケールカテゴリの su の動作が決定します。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数がどれも設定されていなければ、C ロケール (米国の形式) によって su の動作が決定します。</p> <p>LC_CTYPE su が文字を処理する方法を決定します。LC_CTYPE に有効な値が設定されていると、そのロケールで有効な文字を含むテキストやファイル名を表示または処理できます。また、拡張 UNIX コード (EUC) の表示または処理も可能です (1 ~ 3 バイト幅の文字を使用)。このほか、1 カラム幅、2 カラム幅、またはそれ以上のカラム幅の EUC 文字も処理できます。C ロケールでは、ISO 8859-1 の文字だけが有効です。</p> <p>LC_MESSAGES 診断メッセージや情報メッセージの表示方法を決定します。また、メッセージの言語とスタイル、肯定・否定の応答形式も決定します。C ロケールでは、メッセージはプログラム自身が使用しているデフォルトの形 (通常、米語) で表示されます。</p>
ファイル	<pre>\$HOME/.profile sh および ksh ユーザーのログインコマンド /etc/passwd システムのパスワードファイル /etc/profile システム全体の sh と ksh のログインコマンド /var/adm/sulog ログファイル</pre>

su(1M)

/etc/default/su

このファイル中に指定できるデフォルトパラメータは以下のとおりです。

- SULOG 定義されている場合、別のユーザーになるための su の試行はすべて、指定されたファイルに記録されます。
- CONSOLE 定義されている場合、root になるための su の試行はすべて、コンソールに記録されます。
- PATH デフォルトパス (/usr/bin:)
- SUPATH root になるために su を起動しているユーザーのデフォルトパス (/usr/sbin:/usr/bin)
- SYSLOG すべての su 試行を記録するのに syslog(3C) の LOG_AUTH を使うかどうかを指定します。
LOG_NOTICE メッセージは root への su に対して生成され、
LOG_INFO メッセージはその他のユーザーへの su に対して生成されます。LOG_CRIT メッセージは、su に失敗した場合に生成されません。
- SLEEPTIME 存在していれば、ログインに失敗したことが画面上に表示され、次のログインが入力できるまでの待ち時間を秒単位で設定します。デフォルト値は 4、最小値は 0、最大値は 5 です。

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWcsu

関連項目 csh(1)、env(1)、ksh(1)、login(1)、roles(1)、sh(1)、syslogd(1M)、settaskid(2)、pam(3PAM)、syslog(3C)、pam.conf(4)、passwd(4)、profile(4)、sulog(4)、attributes(5)、environ(5)、pam_unix(5)

名前 afbconfig, SUNWafb_config – AFB グラフィックスアクセラレータの設定

形式 **/usr/sbin/afbconfig** [-dev *device-filename*] [-res *video-mode* [now | try]
 [noconfirm | nocheck]] [-file machine | system]
 [-deflinear true | false] [-defoverlay true | false]
 [-overlayorder first | last] [-expvis enable | disable]
 [-sov enable | disable] [-maxwinds *n*]
 [-extovl enable | disable] [-g *gamma-correction-value*]
 [-gfile *gamma-correction-file*] [-propt] [-prconf] [-defaults]

/usr/sbin/afbconfig [-propt] [-prconf]

/usr/sbin/afbconfig [-help] [-res ?]

機能説明 afbconfig は、AFB グラフィックスアクセラレータおよび AFB 対応の X11 ウィンドウシステムのデフォルトの一部を設定します。

afbconfig の次の形式では、指定したオプションを OWconfig ファイルに保存します。

/usr/sbin/afbconfig [-dev*device-filename*] [-res *video-mode* [now | try]
 [noconfirm | nocheck]] [-file machine | system]
 [-deflinear true | false] [-defoverlay true | false]
 [-overlayorderfirst | last] [-expvisenable | disable]
 [-sov enable | disable] [-maxwinds*n*]
 [-extovl enable | disable] [-g*gamma-correction-value*]
 [-gfile*gamma-correction-file*] [-propt] [-prconf] [-defaults]

これらのオプションは、次にウィンドウシステムをそのデバイスで実行するときに AFB デバイスを初期化するために使用されます。OWconfig ファイル内のオプションの更新は、異なるウィンドウセッションや再起動したシステムでも有効となります。

-prconf、-propt、-help、-res ? オプションだけを起動する次の形式では、OWconfig ファイルは更新されません。

/usr/sbin/afbconfig [-propt] [-prconf]

/usr/sbin/afbconfig [-help] [-res ?]

また、次の形式では、その他のオプションはすべて無視されます。

/usr/sbin/afbconfig [-help] [-res ?]

オプションは、一度に 1 つの AFB デバイスに対してのみ指定することができます。複数の AFB デバイスに対してオプションを指定するには、afbconfig コマンドを複数回起動する必要があります。

afbconfig で指定できるのは、AFB 固有のオプションだけです。デフォルトの表示色数、デフォルトのビジュアルクラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください。

SUNWafb_config(1M)

ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは、/usr/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/etc/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。

これらの標準 OWconfig ファイルのどちらもスーパーユーザーのみが書き込みを行います。したがって、スーパーユーザーが所有する afbconfig プログラムは、setuid による root の権限で実行されます。

オプションのデフォルト値

afbconfig コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。ウィンドウシステムを実行する際に、afbconfig による AFB オプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルトは次のとおりです。

-dev	/dev/fbs/afb0
-file	machine
-res	none
-deflinear	false
-defoverlay	false
-linearorder	last
-overlayorder	last
-expvis	enabled
-sov	enabled
-maxwids	32
-extovl	enabled
-g	2.22

-res オプションのデフォルト値 none とは、ウィンドウシステムが実行された場合に、画面解像度がそのデバイスに現在プログラムされている表示モードになることを意味しています。

これによって、PROM によってデバイスの解像度を指定しているユーザーとの共用性が保てます。(Gx などの)一部のデバイスでは、PROM が表示モードを指定する唯一の手段です。これは、デフォルトの AFB 表示モードは、最終的に PROM によって決まることを意味しています。

オプション

次のオプションがサポートされています。

-defaults
すべてのオプションの値をそれぞれのデフォルト値に戻します。

-deflinear true | false

AFB には、2 種類の画像表示形式があります。リニア画像と非リニア画像です。リニア画像はガンマ補正され、非リニア画像は補正されません。リニア画像版も非リニア画像版も、ともに持つ画像表示形式が 2 つあります。24 ビットトゥルーカラーと 8 ビットスタティックグレーです。

true のときは、デフォルトの画像表示形式として、デフォルトで選択されたオプション (特に、Xsun(1) の defdepth および defclass オプション。詳細は OpenWindows のマニュアルページを参照) を満たすリニア画像がセットされます。

false のとき、または他のデフォルトで選択されたオプションを満たすリニア画像がないときは、これらの他のオプションを満たす、非リニア画像がデフォルトとして選択されます。AFB にはリニアオーバーレイ画像表示形式がないため、-defoverlay オプションが存在するときには、このオプションを使用することはできません。

-defoverlay true | false

AFB が、残りの AFB 画像から切り離されたピクセルを持つ 8 ビット疑似カラー画像を提供します。これを、オーバーレイ画像といいます。この画像表示形式で作成されたウィンドウは、他の画像表示形式で作成されたウィンドウに影響を与えません。逆に、他の画像表示形式で作成されたウィンドウは、オーバーレイウィンドウに影響を与えます。

この画像を使用して作成されたウィンドウで使用することができる色の数は -extov1 オプションの設定に依存します。-extov1 オプションが有効になっている場合は、256 種類の不透明カラーの値による拡張オーバーレイを使用することができます(-extov1 を参照)。-extov1 が無効になっている場合は、拡張オーバーレイを使用することはできず、この画像には、不透明カラーの (256 - maxwids) の値が使用されます (-maxwids を参照)。

-defoverlay の値が true である場合には、オーバーレイ画像がデフォルト画像になります。-defoverlay の値が false の場合には、他のデフォルトで選択された def、depth および defclass オプションを満たすオーバーレイでない画像表示形式が、デフォルトの画像表示形式として選択されます。詳細は、OpenWindows のマニュアルページを参照してください。

-defoverlay true オプションが使用されるときは、常に openwin コマンド行で選択されたデフォルトの深さとクラスは 8 ビット疑似カラーである必要があります。それ以外の場合は、警告メッセージが出力され、-defoverlay オプションは false として扱われます。

-deflinear オプションが存在するときには、AFB にはリニアオーバーレイ画像表示形式がないため、このオプションは使用することができません。

-dev device-filename

AFB 特殊ファイルを指定します。デフォルトは /dev/fbs/afb0 です。

SUNWafb_config(1M)

-expvis enable | disable

enable にすると、OpenGL Visual Expansion が起動されます。選択された画像表示形式グループ (8 ビット PseudoColor、24 ビット TrueColor など) が画像表示形式リストに見つかります。

-extovl enable | disable

enable にすると、拡張オーバーレイを使用することができます。このオーバーレイ画像には 256 種類の不透明カラーがあります。SOV 画像には 255 種類の不透明カラーと 1 種類の透明カラーがあります。

また、このオプションは、ハードウェアによる透明カラーを有効にするため、SOV 画像を使用するウィンドウで、より高い性能が得られます。

-file machine | system

更新する OWconfig ファイルを指定します。machine が指定された場合は、/etc/openwin ディレクトリツリーにあるマシン固有の OWconfig ファイルが使用されます。system が指定された場合は、/usr/openwin ディレクトリツリーにある共通の OWconfig ファイルが使用されます。指定されたファイルがない場合は、新たに生成されます。

-g gamma-correction-value

ガンマ補正の値を変更することができます。すべてのリニア画像ではガンマ補正を使用することができます。デフォルトでは、*gamma-correction-value* は 2.22 です。0 より小さい値は無効 (不正) です。ガンマ補正の値はリニア画像に適用され、リニア画像の有効ガンマ値は 1.0 になります。これは、XSolarisGetVisualGamma(3) によって返される値です。この機能については、XSolarisGetVisualGamma(3) を参照してください。

このオプションは、ウィンドウシステムが稼動しているときに使用することができます。ガンマ補正の値を変更すると、リニア画像を使用して表示されているすべてのウィンドウに影響を受けます。

-gfile gamma-correction-file

指定されたファイル (*gamma-correction-file*) からガンマ補正表を読み込みます。*gamma-correction-file* は、各行が R、G、B チャネルの値を持つように書式化されている必要があります。それらの値は、16 進数で指定し、値と値の間は 1 つ以上のスペースで区切ります。また、*gamma-correction-file* は、そのような 3 つの値の組が 256 種類定義されます。

gamma-correction-file の例を以下に示します。

```
0x00 0x00 0x00
0x01 0x01 0x01
0x02 0x02 0x02
...
...
0xff 0xff 0xff
```

このオプションを使用することによって、ウィンドウシステムが稼動しているときにガンマ補正表を読み込むことができます。新しいガンマ補正は、このリニア画像によって表示されているすべてのウィンドウに影響を与えます。ユーザーが指定した表によってガンマ補正を行う際は、ガンマ補正の値は定義されません。デフォルト

トでは、ウィンドウシステムはガンマ補正值として 2.22 を使用し、このガンマ補正值に対応してウィンドウシステムが作成したガンマ補正表を読み込みます。

-help

afbconfig コマンド行のオプションと機能の概要を一覧で表示します。

-linearorder first | last

first のときには、AFB スクリーン用の X11 スクリーン画像表示形式リスト上で、リニア画像が非リニア画像より前に表示されます。last のときには、非リニア画像は、リニア画像より前に表示されます。

-maxwids *n*

ウィンドウ ID s (WIDs) として使用するために予約される最大数の AFB X チャネルピクセル値を指定します。オーバーレイカラーマップのピクセル値の残りは、通常の X11 の未使用のカラーピクセルのために使用されます。確保された WIDs は、(XGL などの) 3 次元グラフィックスウィンドウ、MBX ウィンドウと、デフォルト以外の画像表示形式をもつウィンドウにより発生順に割り当てられます。X チャネルコードの 0 から (255 - *n*) は、未使用のカラーピクセルです。(255 - *n* + 1) から 255 の X チャネルコードは、WID として使用するために予約されます。適切な値は、1、2、4、8、16、32、64 です。

このオプションは -extovl が無効になっている場合のみ使用することができます。

-overlayorder first | last

first のときには、AFB スクリーン用の X11 スクリーン画像表示形式リスト上で、8 ビット疑似カラーオーバーレイ画像が、非オーバーレイ画像より前に表示されます。last のときには、非オーバーレイ画像は、オーバーレイ画像より前に表示されます。

-propt

-file オプションで指定された OWconfig ファイルに書かれた AFB オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。afbconfig の呼び出しが終了した後に、OWconfig ファイルに書き込まれるオプションの値を表示します。

次に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/afb0 ---
OWconfig: machine
Video Mode: 1280x1024x76
Default Visual: Non-Linear Normal Visual
Visual Ordering: Linear Visuals are last
                  Overlay Visuals are last
OpenGL Visual Expansion: enabled
Server Overlay Visuals: enabled
Extended Overlay: enabled
Underlay WIDs: 64 (not configurable)
Overlay WIDs: 4 (not configurable)
Gamma Correction Value: 2.220
Gamma Correction Table: Available
```

SUNWafb_config(1M)

-prconf

AFB ハードウェア構成を表示します。

次に表示例を示します。

```
--- Hardware Configuration for /dev/fbs/afb0 ---
Type: double-buffered AFB with Z-buffer
Board: rev 0 (Horizontal)
Number of Floats: 6
PROM Information: @(#)afb.fth x.xx xx/xx/xx
AFB ID: 0x101df06d
DAC: Brooktree 9070, version 1 (Pac2)
3DRAM: Mitsubishi 130a, version x
EDID Data: Available - EDID version 1 revision x
Monitor Sense ID: 4 (Sun 37x29cm RGB color monitor)
Monitor possible resolutions: 1024x768x77, 1024x800x84, 1
                             1152x900x76, 1280x1024x67, 1280x1024x76, 960x680xx108s
Current resolution setting: 1280x1024x76
```

-sov enable | disable

enable にすると、ルートウィンドウのSERVER_OVERLAY_VISUALS 属性が有効になります。SOV 画像が転送され、それらの透過タイプ、値、階層は、この属性によって参照することができます。disable にすると、SERVER_OVERLAY_VISUALS 属性は定義されません。SOV 画像は転送されません。

-res *video-mode* [now | try [noconfirm | nocheck]]

指定した AFB デバイスに接続されているモニターを制御する際に使われる表示モードを指定します。

組み込まれている表示モードの形式は次のとおりです。 *widthxheightxrate* *width* はピクセル単位のスクリーン幅、*height* はピクセル単位のスクリーンの高さ、*rate* は画面を垂直方向に再描画する周期です。

960x680x112s や 960x680x108s の s 接尾辞は、これらが立体表示モードであることを意味します。640x480x60i や 768x575x50i の i 接尾辞は、インタレース表示タイミングを有効にします。この接尾辞がない場合は、ノンインタレースタイミングが使用されます。

便宜上、-res にリフレッシュレートを指定する際、値の直前に x の代わりに @ を使用できます。たとえば、1280x1024@76 のように指定できます。AFB が対応している一部の表示モードには、モニターが対応していない場合があります。また、AFB がサポートする表示モードにも、モニターがサポートしていないものがあります。AFB デバイスとモニターの両方がサポートしている表示モードのリストは、-res ? オプション付きの afbconfig (形式の項に記された 3 番目の形式) を実行することによって得ることができます。

AFB がサポートしている表示モードのリストを次に示します。

```
1024x768x60
1024x768x70
1024x768x75
```

```

1024x768x77
1024x800x84
1152x900x66
1152x900x76
1280x800x76
1280x1024x60
1280x1024x67
1280x1024x76
960x680x112s (立体表示)
960x680x108s (立体表示)
640x480x60
640x480x60i (インタレース)
768x575x50i (インタレース)

```

便宜上、AFB がサポートしている表示モードのいくつかには記号名が定義されています。*widthxheightxrate* の形式の代わりに、記号名を `-res` オプションの引数として指定することができます。記号名 `none` は、ウィンドウシステムを実行すると、画面の解像度は現在デバイスにプログラムされている表示モードになることを意味します。

AFB がサポートしている表示モードのリストを次に示します。

記号名 対応する表示モード

```

svga 1024x768x60
1152 1152x900x76
1280 1280x1024x76
stereo 960x680x112s
ntsc 640x480x60i
pal 768x575x50i
none (上記参照)

```

`-res` オプションには、表示モードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定できます。

`noconfirm` `-res` オプションを指定した際に、システムが使用不可であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために `afbconfig` のデフォルトの動作では、この問題についての警告メッセージと、処理を継続するかどうかを確認するメッセージを表示します。`noconfirm` オプションを指定すると、`afbconfig` コマンドはこの確認をせずに、要求のあった表示モードにプログラムします。このオプションは、`afbconfig` がシェルスクリプトから実行されている場合に便利です。

`nocheck` このオプションを指定すると、モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定された表示モードは、現在接続されているモニターに適切かど

SUNWafb_config(1M)

	<p>うかにかかわらず受け付けられます。このオプションは、AFB デバイスに異なるモニターを接続する場合に便利です。このオプションの指定は、noconfirm の指定も兼ねます。</p>				
now	<p>OWconfig ファイルの表示モードを更新するとともに、AFB デバイスが指定した表示モードにただちにプログラムされます。この機能は、ウィンドウシステムを開始する前に表示モードを変更する際に便利です。</p> <p>対象となるデバイスが稼働している間 (たとえば、ウィンドウシステムの稼働中) に、この引数をafbconfig に指定することはお勧めしません。予期しない結果になることもあります。now 引数を指定してafbconfig コマンドを実行する場合は、最初にウィンドウシステムを終了してください。now 引数がウィンドウシステムのセッション中に使用された場合、表示モードはただちに変更されますが、画面の幅や高さはそのセッションが終了して次のセッションに入るまで変更されません。さらに、立体表示モードではシステムが変更を認識しないことがあります。したがって、ウィンドウシステムの稼働中には絶対に now オプションを指定しないでください。</p>				
try	<p>このオプションを指定すると、指定した表示モードが試験的にプログラムされます。ユーザーは、指定した表示モードを使用する場合は、メッセージが表示されてから 10 秒以内に「y」を入力します。表示されたモードを使用しない場合は、10 秒以内に任意の文字を入力します。「y」またはReturn キー以外の文字の入力は、すべて「使用しない」と判断され、以前の表示モードに戻されるため、OWconfig ファイル中の表示モードは書き換えられません (その他の指定されたオプションは有効となります)。Return キーの入力があった場合は、新しい表示モードを保持するかどうかを yes または no で確認するメッセージが表示されます。このオプションの指定は、now 引数の指定も兼ねます (now 引数の注意を参照)。</p>				
使用例	<p>例 1 モニターの種類の変更</p> <p>モニターの種類を、垂直周波数 76 Hz で解像度 1280x1024 に変更する例を以下に示します。</p> <pre>example% /usr/sbin/afbconfig -res 1280x1024x76</pre>				
属性	<p>次の属性については、attributes(5) を参照してください。</p>				
	<table border="1"> <thead> <tr> <th>属性タイプ</th> <th>属性値</th> </tr> </thead> <tbody> <tr> <td>使用条件</td> <td>SUNWafbconfig</td> </tr> </tbody> </table>	属性タイプ	属性値	使用条件	SUNWafbconfig
属性タイプ	属性値				
使用条件	SUNWafbconfig				

SUNWafb_config(1M)

関連項目 | mmap(2), attributes(5)

SUNWffb_config(1M)

名前	ffbconfig, SUNWffb_config – FFB グラフィックスアクセラレータの設定
形式	<pre> /usr/sbin/ffbconfig [-dev device-filename] [-res video-mode [now try] [noconfirm nocheck]] [-file machine system] [-deflinear true false] [-defoverlay true false] [-linearorder first last] [-overlayorder first last] [-expvis enable disable] [-sov enable disable] [-maxwids n] [-extovl enable disable] [-g gamma-correction-value] [-gfile gamma-correction-file] [-propt] [-prconf] [-defaults] /usr/sbin/ffbconfig [-propt] [-prconf] /usr/sbin/ffbconfig [-help] [-res ?] </pre>
機能説明	<p>ffbconfigは、FFB グラフィックスアクセラレータおよび FFB 対応の X11 ウィンドウシステムのデフォルトの一部を設定します。</p> <p>ffbconfigの1番目の形式では、指定したオプションを OWconfigファイルに保存します。これらのオプションは、次にウィンドウシステムをそのデバイスで実行するときに FFB デバイスを初期化するために使用されます。OWconfig ファイル内のオプションの更新は、異なるウィンドウセッションや再起動したシステムでも有効となります。</p> <p>-prconf、-propt、-help、-res? オプションだけを起動する2番目と3番目の形式では、OWconfig ファイルは更新されません。また、3番目の形式では、その他のオプションはすべて無視されます。</p> <p>オプションは、一度に1つの FFB デバイスに対してのみ指定することができます。複数の FFB デバイスに対してオプションを指定するには、ffbconfig を複数回起動する必要があります。</p> <p>ffbconfig で指定できるのは、FFB 固有のオプションだけです。デフォルトの表示色数、デフォルトの画像表示形式クラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください。詳細については、OpenWindows デスクトップ・リファレンスマニュアルを参照してください。</p> <p>ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは、/etc/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/usr/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。</p> <p>これらの標準 OWconfig ファイルのどちらもスーパーユーザーのみが書き込みを行います。したがって、スーパーユーザーが所有する ffbconfig プログラムは、setuid による root の権限で実行されます。</p>
オプション	<pre> -dev device-filename FFB 特殊ファイルを指定します。デフォルトは/dev/fbs/ffb0 です。 -file machine system 更新する OWconfig ファイルを指定します。machine を指定すると、 /etc/openwin ディレクトリツリーにあるマシン固有の OWconfig ファイルが使 </pre>

用されます。system を指定すると、/usr/openwin ディレクトリツリーにある共通のOWconfig ファイルが使用されます。ファイルがない場合は、新たに生成されます。

`-res video-mode [now | try [noconfirm | nocheck]]`

指定した FFB デバイスに接続されているモニターを制御する際に使われる表示モードを指定します。

表示モードの形式は *widthxheightxrate* で、*width* はピクセル単位の画面幅、*height* はピクセル単位の画面の高さ、*rate* は画面を垂直方向に再描画する周期です。

960x680x112s や 960x680x108s の s 接尾辞は、これらが立体表示モードであることを意味します。640x480x60i や 768x575x50i の i 接尾辞は、インタレース表示タイミングを有効にします。この接尾辞がない場合は、ノンインタレースタイミングが使用されます。

`-res` (「形式」に記されている 3 番目の形式) にリフレッシュレートを指定する際は、値の直前に x の代わりに @ を使用することができます。たとえば、1280x1024@76 のように指定することができます。

一部の表示モードは、FFB の一部のバージョンのみが対応しています。また、FFB が対応している表示モードには、モニターが対応していないものもあります。FFB デバイスとモニターの両方が対応している表示モードのリストは、`-res ? オプション付きの ffbconfig` を実行することによって得ることができます。

FFB が対応している表示モードのリストを以下に示します。

表示モード	説明
1024x768x60	
1024x768x70	
1024x768x75	
1024x768x77	
1024x800x84	
1152x900x66	
1152x900x76	
1280x800x76	
1280x1024x60	
1280x1024x67	
1280x1024x76	
960x680x112s	(立体表示)

SUNWffb_config(1M)

表示モード	説明
960x680x108s	(立体表示)
640x480x60	
640x480x60i	(インタレース)
768x575x50i	(インタレース)
1440x900x76	(高解像度)
1600x1000x66	(高解像度)
1600x1000x76i	(高解像度)
1600x1280x76	(高解像度)
1920x1080x72	(高解像度)
1920x1200x70	(高解像度)

記号名

便宜上、表示モードのいくつかには記号名が定義されています。 *widthxheightxrate* の形式の代わりに、記号名を `-res` の引数として指定することができます。記号名 `none` は、ウィンドウシステムを実行すると、画面の解像度は現在デバイスにプログラムされている表示モードになることを意味します。

記号名	対応する表示モード
svga	1024x768x60
1152	1152x900x76
1280	1280x1024x76
stereo	960x680x112s
ntsc	640x480x60i
pal	768x575x50i
none	(デバイスでプログラムされている表示モード)

`-res` オプションには、表示モードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定することができます。

now

OWconfig ファイルの表示モードを更新するとともに、FFB デバイスが指定した表示モードにただちにプログラムされます (この機能は、ウィンドウシステムを開始する前に表示モードを変更する際に便利です)。

対象となるデバイスが稼働している間 (たとえば、ウィンドウシステムの稼働中) に、この追加オプションを `ffbconfig` に指定することはお勧めしません。予期しない結果になることもあります。 `now` オプションを指定して `ffbconfig` コマンドを実行する場合は、最初にウィンドウシステムを終了してください。 `now` オプションがウィンドウシステムのセッション中に使用された場合、表示モードはただちに変更されますが、画面の幅や高さはそのセッションが終了して次のセッションに入るまで変更されません。さらに、立体表示モードではシステムが変更を認識しないことがあります。したがって、ウィンドウシステムの稼働中には絶対に `now` オプションを指定しないでください。

`noconfirm`

確認と警告メッセージを省略し、要求された表示モードにプログラムします。

`-res` オプションを指定した際に、システムが使用可能であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために `ffbconfig` のデフォルトの動作では、この問題についての警告メッセージと処理を継続するかどうかを確認するメッセージを表示します。このオプションは、`ffbconfig` がシェルスクリプトから実行されている場合に便利です。

`nocheck`

モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定された表示モードは、現在接続されているモニターに適切かどうかにかかわらず受け付けられます。このオプションは、FFB デバイスに異なるモニターを接続する場合に便利です。このオプションを指定すると、`noconfirm` も指定されます。

`try`

指定した表示モードに試験的にプログラムされます。ユーザーは、指定した表示モードを使用する場合は、メッセージが表示されてから 10 秒以内に `y` と入力します。表示されたモードを使用しない場合は、10 秒以内に任意の文字を入力します。 `y` または `Return` キー以外の文字の入力は、すべて「使用しない」とみなされ、以前の表示モードに戻され、`OWconfig` ファイル中の表示モードは書き換えられません。その他の指定されたオプションは有効となります。 `Return` キーの入力があつた場合は、新しい表示モードを保持するかどうかを `yes` または `no` で確認するメッセージが表示されます。このオプションを指定すると、`now` サブオプションも指定されます。(`now` サブオプションの注意を参照)。

`-deflinear true | false`

FFB には、2 種類の画像表示形式があります。リニア画像と非リニア画像です。リニア画像はガンマ補正され、非リニア画像は補正されません。リニア画像版も非リニア画像版も、ともに持つ画像表示形式が 2 つあります。24 ビット `TrueColor` と 8 ビット `StaticGray` です。

`-deflinear true` を指定すると、デフォルトの画像表示形式として、デフォルトで選択されたオプション (特に、Xsun(1)の `defdepth` および `defclass` オプション。詳細は *OpenWindows* デスクトップ・リファレンスマニュアルのマニュアルページを参照) を満たすリニア画像を設定します。具体的には、デフォルトの画像表示形式の選択オプションは、Xsun(1)の `defdepth` および `defclass` オプションによって設定されたものです。

`-deflinear false` を指定すると、または他のデフォルトで選択されたオプションを満たすリニア画像がない場合は、これらの他のオプションを満たす非リニア画像がデフォルトとして選択されます。

FFB にはリニアオーバーレイ画像表示形式がないため、オプションが存在する場合は、このオプションを使用することはできません。

`-defoverlay true | false`

FFB が、残りの FFB 画像から切り離されたピクセルを持つ 8 ビット疑似カラー画像を提供します。これを、オーバーレイ画像といいます。この画像表示形式で作成されたウィンドウは、他の画像表示形式で作成されたウィンドウに影響を与えません。逆に、他の画像表示形式で作成されたウィンドウは、オーバーレイウィンドウに影響を与えます。この画像表示形式では、256 種類の不透明カラーの値による拡張オーバーレイを使用することができます。 `-maxwids` を参照してください。

`-defoverlay` に `true` を指定すると、オーバーレイ画像がデフォルト画像になります。 `-defoverlay` に `false` を指定すると、他のデフォルトで選択された `defdepth` および `defclass` オプションを満たすオーバーレイでない画像表示形式が、デフォルトの画像表示形式として選択されます。詳細は、*OpenWindows* デスクトップ・リファレンスマニュアルのマニュアルページを参照してください。

`-defoverlay true` を使用する場合は、`openwin` コマンド行で選択されたデフォルトの深さとクラスは常に 8 ビット疑似カラーである必要があります。それ以外の場合は、警告メッセージが出力され、`-defoverlay` オプションは `false` として扱われます。 `-deflinear` オプションが存在する場合、FFB にはリニアオーバーレイ画像表示形式がないため、このオプションは使用することができません。

`-linearorder first | last`

`first` を指定すると、FFB 画面用の X11 画面画像表示形式リスト上で、リニア画像が非リニア画像より前に表示されます。 `last` を指定すると、非リニア画像は、リニア画像より前に表示されます。

`-overlayorder first | last`

`first` を指定すると、FFB 画面用の X11 画面画像表示形式リスト上で、8 ビット疑似カラーオーバーレイ画像が、非オーバーレイ画像より前に表示されます。 `last` を指定すると、非オーバーレイ画像は、オーバーレイ画像より前に表示されます。

`-expvis enable | disable`

`enable` を指定すると、OpenGL Visual Expansion が起動されます。選択された画像表示形式グループ (8 ビット `PseudoColor`、24 ビット `TrueColor` など) は、画面画像表示形式リストで見つけることができます。

-sov enable | disable

enable を指定すると、ルートウィンドウの `SERVER_OVERLAY_VISUALS` 属性が有効になります。SOV 画像が転送され、それらの透過タイプ、値、階層は、この属性によって参照することができます。disable を指定すると、`SERVER_OVERLAY_VISUALS` 属性は定義されません。SOV 画像は転送されません。

-maxwids *n*

ウィンドウ ID (WID) として使用するために予約される最大数の FFB チャンネルピクセル値を指定します。オーバーレイカラーマップのピクセル値の残りは、通常の X11 の未使用のカラーピクセルのために使用されます。確保された WID は、(XGL などの) 3次元グラフィックスウィンドウ、MBX ウィンドウと、デフォルト以外の画像表示形式をもつウィンドウにより発生順に割り当てられます。X チャンネルコードの 0 から (255-*n*) は、未使用のカラーピクセルです。(255-*n*+1) から 255 の X チャンネルコードは、WID として使用するために予約されます。FFB と FFB2 の有効な値は、1、2、4、8、16、32 です。FFB2+ の有効な値は、1、2、4、8、16、32、64 です。

-extovl enable | disable

このオプションは FFB2+ のみで使用することができます。enable を指定すると、拡張オーバーレイを使用することができます。このオーバーレイ画像には 256 種類の不透明カラーがあります。SOV 画像には 255 種類の不透明カラーと 1 種類の透明カラーがあります。このオプションは、ハードウェアによる透明カラーを有効にするため、SOV 画像を使用するウィンドウで、より高い性能が得られます。

-g *gamma-correction value*

このオプションは FFB2+ のみで使用することができます。このオプションによって、ガンマ補正の値を変えることができます。すべてのリニア画像ではガンマ補正を使用することができます。デフォルトでは、ガンマ補正の値は 2.22 です。0 より小さい値は無効 (不正) です。ガンマ補正の値はリニア画像に適用され、リニア画像の有効ガンマ値は 1.0 になります。これは、`XSolarisGetVisualGamma(3)` によって返される値です。この機能については、`XSolarisGetVisualGamma(3)` を参照してください。

このオプションは、ウィンドウシステムが稼働しているときに使用することができます。ガンマ補正の値を変更すると、リニア画像を使用して表示されているすべてのウィンドウが影響を受けます。

-gfile *gamma-correction file*

このオプションは FFB2+ のみで使用することができます。このオプションは、指定されたファイルからガンマ補正表を読み込みます。このファイルは、各行が R、G、B チャンネルの値を持つように書式化されている必要があります。それらの値は、16 進数で指定し、値と値の間は 1 つ以上の空白文字で区切ります。このファイルでは、3 つの値の組が 256 種類定義されます。このファイルの例を以下に示します。

```
0x00 0x00 0x00
0x01 0x01 0x01
0x02 0x02 0x02
...
...
```

SUNWffb_config(1M)

```
0xff 0xff 0xff
```

このオプションを使用することによって、ウィンドウシステムが稼動しているときにガンマ補正表を読み込むことができます。新しいガンマ補正は、このリニア画像によって表示されているすべてのウィンドウに影響を与えます。ユーザーが指定した表によってガンマ補正を行う際は、ガンマ補正の値は定義されません。デフォルトでは、ウィンドウシステムはガンマ補正值として2.22を使用し、このガンマ補正值に対応してウィンドウシステムが作成したガンマ補正表を読み込みます。

-defaults

すべてのオプションの値をそれぞれのデフォルト値に戻します。

-propt

-file オプションで指定された OWconfig ファイルに書かれた FFB オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。ffbconfig の呼び出しが終了した後に、OWconfig ファイルに書き込まれるオプションの値を表示します。次に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/ffb0 ---
OWconfig: machine
Video Mode: NONE
Default Visual: Non-Linear Normal Visual
Visual Ordering: Linear Visuals are last
                  Overlay Visuals are last
OpenGL Visuals: disabled
SOV: disabled
Allocated WIDs: 32
```

-prconf

FFB のハードウェア構成を表示します。次に表示例を示します。

```
--- Hardware Configuration for /dev/fbs/ffb0 ---
Type: double-buffered FFB2 with Z-buffer
Board: rev x
PROM Information: @(#)ffb2.fth x.x xx/xx/xx
FBC: version x
DAC: Brooktree 9068, version x
3DRAM: Mitsubishi 1309, version x
EDID Data: Available - EDID version 1 revision x
Monitor Sense ID: 4 (Sun 37x29cm RGB color monitor)
Monitor possible resolutions: 1024x768x60, 1024x768x70,
                              1024x768x75, 1152x900x66, 1152x900x76,
                              1280x1024x67, 1280x1024x76, 960x680x112s,
                              640x480x60
Current resolution setting: 1280x1024x76
```

-help

ffbconfig コマンド行のオプションと機能の概要を一覧で表示します。

デフォルト設定

ffbconfig コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。

SUNWffb_config(1M)

ウィンドウシステムを実行する際に、ffbconfig による FFB オプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルト値を以下に示します。

オプション	デフォルト値
-dev	/dev/fbs/ffb0
-file	machine
-res	none
-deflinear	false
-defoverlay	false
-linearorder	last
-overlayorder	last
-expvis	enabled
-sov	enabled
-maxwids	32

-res オプションのデフォルト値 none とは、ウィンドウシステムが実行された場合に、画面解像度がそのデバイスに現在プログラムされている表示モードになることを意味しています。

注：これによって、PROM によってデバイスの解像度を指定しているユーザーとの共用性が保てます。(gx などの)一部のデバイスでは、PROM が表示モードを指定する唯一の手段です。これは、デフォルトの FFB 表示モードは、最終的に PROM によって決まることを意味しています。

使用例 例 1 モニターの種類の変更

モニターの種類を、垂直周波数 76 Hz で解像度 1280 × 1024 に変更する例を以下に示します。

```
example% /usr/sbin/ffbconfig -res 1280x1024x76
```

ファイル /dev/fbs/ffb0 デバイス特殊ファイル

属性 次の属性については、attributes(5) を参照してください。

属性タイプ	属性値
使用条件	SUNWffbcf

関連項目 mmap(2), attributes(5), fbio(7I), ffb(7D)

SUNWffb_config(1M)

OpenWindows デスクトップ・リファレンスマニュアルのマニュアルページ

名前	m64config, SUNWm64_config – M64 グラフィックスアクセラレータの設定
形式	<pre> /usr/sbin/m64config [-defaults] [-depth 8 24] [-dev device-filename] [-file machine system] [-prconf] [-propt] [-res video-mode now try] [noconfirm nocheck] /usr/sbin/m64config [-prconf] [-propt] /usr/sbin/m64config [-help] [-res ?] </pre>
機能説明	<p>m64config は、M64 グラフィックスアクセラレータおよび M64 対応の X11 ウィンドウシステムのデフォルトの一部を設定します。</p> <p>形式の項に記された m64config の 1 番目の形式では、指定したオプションを OWconfig ファイルに保存します。これらのオプションは、次にウィンドウシステムをそのデバイスで実行するときに M64 デバイスを初期化するために使用されます。OWconfig ファイル内のオプションの更新は、異なるウィンドウセッションや再起動したシステムでも有効となります。</p> <p>-prconf、-propt、-help、-res ? オプションだけを起動する 2 番目と 3 番目の形式では、OWconfig ファイルは更新されません。また、3 番目の形式では、その他のオプションはすべて無視されます。</p> <p>オプションは、一度に 1 つの M64 デバイスに対してのみ指定することができます。複数の M64 デバイスに対してオプションを指定するには、m64config を複数回起動する必要があります。</p> <p>m64config で指定できるのは、M64 固有のオプションだけです。デフォルトの表示色数、デフォルトの画像表示形式クラスなどを指定する通常のウィンドウシステムのオプションは、openwin コマンド行のデバイス修飾子で指定してください (OpenWindows のマニュアルページの Xsun(1) を参照)。</p> <p>ユーザーは、更新する OWconfig ファイルを指定することもできます。デフォルトでは、/etc/openwin ディレクトリツリーにあるマシン固有のファイルが更新されます。別のファイルを指定するには、-file オプションを使用します。たとえば、/usr/openwin ディレクトリツリーにあるシステム共通の OWconfig ファイルを代わりに更新することができます。</p> <p>これらの標準 OWconfig ファイルのどちらもスーパーユーザーのみが書き込みを行います。したがって、スーパーユーザーが所有する m64config プログラムは、setuid による root の権限で実行されます。</p>
オプション	<p>-defaults すべてのオプションの値をそれぞれのデフォルト値に戻します。</p> <p>-depth 8 24 表示色数を 8 または 24 ビット/ピクセルで指定します。変更を反映させるには、現在のウィンドウシステムのセッションからログアウトしてから再度ログインします。24 ビット/ピクセルでは、画面解像度を低くすることにより、ウィンドウシステムで TrueColor グラフィックスを実現できます。</p>

SUNWm64_config(1M)

24 ビット/ピクセルで利用できる解像度の最大値は、PGX カードに搭載されているメモリー量に依存します。2 M バイトのメモリーを搭載した PGX カードで利用できる解像度の最大値は 800x600 です。4 M バイトのメモリーを搭載したカードで利用できる解像度の最大値は 1152x900 です。指定した解像度と色数の組み合わせに必要なメモリーが不足している場合、m64config はエラーメッセージを出力して終了します。

-dev *device-filename*

M64 特殊ファイルを指定します。デフォルトは /dev/fbs/m640 です。

-file *machine|system*

更新する OWconfig ファイルを指定します。machine を指定すると、/etc/openwin ディレクトリツリーにあるマシン固有の OWconfig ファイルが使用されます。system を指定すると、/usr/openwin ディレクトリツリーにある共通の OWconfig ファイルが使用されます。ファイルがない場合は、新たに生成されます。

-help

m64config コマンド行のオプションと機能の概要を一覧で表示します。

-prconf

M64 ハードウェア構成を表示します。次に表示例を示します。

```
--- Hardware Configuration for /dev/fbs/m640 ---
ASIC: version 0x41004754
DAC: version 0x0
PROM: version 0x0
Card possible resolutions:  640x480x60, 800x600x75, 1024x768x60
    1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76
    1280x1024x60, 1152x900x66, 1152x900x76, 1280x1024x67
    960x680x112S, 960x680x108S, 640x480x60i, 768x575x50i, 1280x800x76
    1440x900x76, 1600x1000x66, 1600x1000x76, vga, svga, 1152, 1280
    stereo, ntsc, pal
Monitor possible resolutions:  720x400x70, 720x400x88, 640x480x60
    640x480x67, 640x480x72, 640x480x75, 800x600x56, 800x600x60
    800x600x72, 800x600x75, 832x624x75, 1024x768x87, 1024x768x60
    1024x768x70, 1024x768x75, 1280x1024x75, 1280x1024x76, 1152x900x66
    1152x900x76, 1280x1024x67, 960x680x112S, vga, svga, 1152, 1280
    stereo
Possible depths: 8, 24
Current resolution setting: 1280x1024x76
Current depth: 8
```

-propt

-file オプションで指定された OWconfig ファイルに書かれた M64 オプションの値のうち、-dev オプションで指定されたデバイスに対するものすべてを表示します。m64config の呼び出しが終了した後に、OWconfig ファイルに書き込まれるオプションの値を表示します。次に表示例を示します。

```
--- OpenWindows Configuration for /dev/fbs/m640 ---
OWconfig: machine
Video Mode: not set
```

```
-res video-mode [ now | try [ noconfirm | nocheck ] ]
```

指定した M64 デバイスに接続されているモニターを制御する際に使われる表示モードを指定します。表示モードはあらかじめ組み込まれています。表示モードの形式は *widthx-heightxrate* で、*width* はピクセル単位の画面幅、*height* はピクセル単位の画面の高さ、*rate* は画面を垂直方向に再描画する周期です。便宜上、*-res* にリフレッシュレートを指定する際は、値の直前に *x* の代わりに *@* を使用することができます。たとえば、1280x1024@76 のように指定することができます。

有効な表示モードのリストは、`m64config -res '?'` を実行することによって得ることができます。? は引用符で囲んで文字として扱う必要があります。すべてのビデオボードおよびモニターがすべての解像度に対応しているわけではありません。m64config は、ボードが対応していない解像度に設定することを許しません。モニターが対応していない解像度に設定しようとする、確認を促すメッセージを表示します。

記号名

便宜上、上記の表示モードのいくつかには記号名が定義されています。*widthxheightxrate* の形式の代わりに、記号名を *-res* の引数として指定することができます。記号名 *none* は、ウィンドウシステムを実行すると、画面の解像度は現在デバイスにプログラムされている表示モードになることを意味します。

記号名	対応する表示モード
svga	1024x768x60
1152	1152x900x76
1280	1280x1024x76
none	(現在デバイスにプログラムされている表示モード)

-res オプションには、表示モードの直後に次の追加引数を指定することができます。追加引数は、単独でも複数でも指定することができます。

nocheck

このオプションを指定すると、モニターセンスコードに基づく通常のエラーチェックが行われません。ユーザーによって指定された表示モードは、現在接続されているモニターに適切かどうかにかかわらず受け付けられます。このオプションは、M64 デバイスに異なるモニターを接続する場合に便利です。このオプションを指定すると、*noconfirm* も指定されます。

noconfirm

-res オプションを指定した際に、システムが使用可能であっても、表示出力のない状態になる場合があります。このような状況は、特定のコードが読み込まれた際のモニターセンスコードにあいまいさがあった場合などに発生します。このような事態を避けるために *m64config* のデフォルトの動作では、この問題についての警告メッセージと、処理を継続するかどうかを確認するメッセージを表示します。*noconfirm* オプションを指定すると、*m64config* コマンドはこの確認をせずに、

SUNWm64_config(1M)

要求のあった表示モードにプログラムします。このオプションは、m64config がシェルスクリプトから実行されている場合に便利です。

now

OWconfig ファイルの表示モードを更新するとともに、M64 デバイスが指定した表示モードにただちにプログラムされます (この機能は、ウィンドウシステムを開始する前に表示モードを変更する際に便利です)。

対象となるデバイスが稼働している間 (たとえば、ウィンドウシステムの稼働中) に、この追加オプションを m64config に指定することはお勧めしません。予期しない結果になることもあります。now オプションを指定して m64config コマンドを実行する場合は、最初にウィンドウシステムを終了してください。now オプションがウィンドウシステムのセッション中に使用された場合、表示モードはただちに変更されますが、画面の幅や高さはそのセッションが終了して次のセッションに入るまで変更されません。さらに、立体表示モードではシステムが変更を認識しないことがあります。したがって、ウィンドウシステムの稼働中には絶対に now オプションを指定しないでください。

try

このオプションを指定すると、指定した表示モードが試験的にプログラムされます。ユーザーは、指定した表示モードを使用する場合は、メッセージが表示されてから 10 秒以内に y と入力します。表示されたモードを使用しない場合は、10 秒以内に任意の文字を入力します。y または Return キー以外の文字の入力は、すべて「使用しない」とみなされ、以前の表示モードに戻され、OWconfig ファイル中の表示モードは書き換えられません (その他の指定されたオプションは有効となります)。Return キーの入力があつた場合は、新しい表示モードを保持するかどうかを yes または no で確認するメッセージが表示されます。このオプションを指定すると、now サブオプションも指定されます。(now サブオプションの注意を参照)。

デフォルト設定

m64config コマンド行で指定されていないオプションについては、対応する OWconfig ファイル中のオプションは更新されず、ファイル内の値がそのまま使用されます。

ウィンドウシステムを実行する際に、m64config による M64 オプションの指定がまったくなかった場合は、デフォルト値が使用されます。オプションのデフォルトを以下に示します。

オプション	デフォルト値
-dev	/dev/fbs/m640
-file	machine
-res	none

-res オプションのデフォルト値 none とは、ウィンドウシステムが実行された場合に、画面解像度がそのデバイスに現在プログラムされている表示モードになることを意味しています。

SUNWm64_config(1M)

これによって、PROM によってデバイスの解像度を指定しているユーザーとの共用性が保てます。(GX などの) 一部のデバイスでは、PROM が表示モードを指定する唯一の手段です。これは、デフォルトの M64 表示モードは、最終的に PROM によって決まることを意味しています。

使用例 例 1 モニターの種類の変更

モニターの種類を、垂直周波数 76 Hz で解像度 1280x1024 に変更する例を以下に示します。

```
example% /usr/sbin/m64config -res 1280x1024x76
```

ファイル /dev/fbs/m640
デバイス特殊ファイル

/usr/openwin/server/etc/OWconfig
システム設定ファイル

/usr/lib/fbconfig/SUNWm64_config
usr/sbin/m64config へのシンボリックリンク

属性 以下の属性については、attributes(5) を参照してください。

属性タイプ	属性値
使用条件	SUNWm64cf

関連項目 attributes(5), m64(7D)

『OpenWindows デスクトップ・リファレンスマニュアル』

ttymon(1M)

名前	ttymon – 端末ポートのポートモニター
形式	<pre>/usr/lib/saf/ttymon /usr/lib/saf/ttymon -g [-d device] [-h] [-t timeout] [-l ttylabel] [-p prompt] [-m modules] [-T termtype]</pre>
機能説明	<p>ttymon は STREAMS ベースの TTY ポートモニターです。その機能には、ポートの監視、ポートの端末モード、ボーレートおよび回線規約の設定、そしてポートに関連したサービスへのユーザーまたはアプリケーションの接続があります。通常、ttymon は、サービス・アクセス・ファシリティ (SAF) の一部として、サービス・アクセス・コントローラ、sac(1M) の制御下で実行するように構成されています。ttymon は sacadm(1M) コマンドを実行して構成されます。ttymon の各インスタンスは複数のポートを監視できます。ttymon のインスタンスによって監視されるポートは、ポートモニターの管理ファイルに指定されています。管理ファイルは pmadm(1M) コマンドおよび ttyadm(1M) コマンドを実行して構成されます。ttymon のインスタンスは、sac コマンドによって呼び出されると、ポートの監視を始めます。各ポートについて、ttymon は回線規約が指定されていれば、最初に回線規約を初期化してから、速度と端末設定を初期化します。/etc/logindevperm 中のエントリに指定されているポートには、デバイスのオーナー、グループ、アクセス権が設定されます (logindevperm(4) 参照)。初期化に使用される値は、TTY 設定ファイルの適切なエントリから取り込まれます。このファイルは sttydefs(1M) コマンドによって管理されています。通常、ポート上のデフォルトの回線規約は、オートプッシュファシリティの autopush(1M) コマンドによって設定されます。</p> <p>次に ttymon はプロンプトを書き込んでユーザーの入力を待ちます。ユーザーが BREAK キーを押して速度が適切でないことを知らせると、ttymon は次の速度を試行してプロンプトを再び書き込みます。正しい入力を受信すると、ttymon は、(もしあれば) ポートのサービスごとの構成ファイルを解釈し、必要に応じて utmpx エントリを作成します (utmpx(4) 参照)。そして、サービス環境を確立し、ポートに関連するサービスを呼び出します。有効な入力は、キャリッジリターンで終わり、少なくとも 1 つの復帰改行以外の文字が入った文字列からなります。サービスが終了すると、ttymon は utmpx エントリが存在していればそのエントリを消去して、ポートを初期状態に戻します。</p> <p><i>autobaud</i> がポートで動作可能であれば、ttymon は自動的にポートのボーレートを判別しようとします。ユーザーがキャリッジリターンを入力してからでないと、ttymon はボーレートを認識してプロンプトをプリントできません。現在のところ、<i>autobaud</i> が判別できるボーレートは 110、1200、2400、4800、および 9600 です。</p> <p>ポートが双方向性ポートとして構成されている場合、ttymon によって、ユーザーはサービスに接続できるようになり、またポートが使用されていなければ、uucico(1M)、cu(1C)、または ct(1C) はダイアルアウト用にポートを使用できるようになります。ポートが双方向性であれば、ttymon は文字の読み込みが終わるのを待ってからプロンプトをプリントします。</p> <p>ポート用に <i>connect-on-carrier</i> フラグをセットしている場合、ttymon は、接続要求を受け取るとただちにポートに関連付けられたサービスを呼び出します。プロンプトメッセージは送信されません。</p>

サービスの呼び出し	<p>ポートが動作禁止になっていると、ttymonはこのポートにおいてサービスを呼び出しません。動作禁止メッセージが指定されている場合には、ttymonは接続要求を受け取ると動作禁止メッセージを送信します。ttymonが動作禁止になっている場合、ttymonのインスタンスの制御下にあるすべてのポートが動作禁止になります。</p> <p>ポートにttymonが呼び出すサービスは、ttymon管理ファイルに指定されています。ttymonは、このポートに呼び出されるサービスを指示する文字列を走査して、%dまたは%%の2文字のシーケンスを捜します。%dを見つけると、ttymonは、これらの2文字をこのポートの完全パス名(デバイス名)に置き換えることによって、実行されるサービスコマンドを変更します。%%を見つけると、これらは単一の%に置き換えられます。</p>
セキュリティ	<p>サービスを呼び出すと、読み書きするためにポートデバイスに対して0、1、および2の各ファイル記述子がオープンされます。サービスは、ttymonに登録したときに用いたユーザー名についてのユーザーID、グループID、および現在のホームディレクトリを使用して呼び出されます。2つの環境変数、HOMEおよびTTYPROMPTは、ttymonによってサービスの環境に追加されます。HOMEはサービスを呼び出すときに用いるユーザー名のホームディレクトリに設定されています。TTYPROMPTはこのポートのサービスに構成されたプロンプト文字列に設定されています。ttymonが呼び出すサービスが、プロンプトが実際にttymonによって出されていたかを判別し、出されている場合にはそのプロンプトが実際には何であったかを判断する機能を持つように、設定されています。</p> <p>サービスアクセスコントローラの制御下にあるttymonにより監視されるポートに設定できるオプションについては、ttyadm(1M)を参照してください。</p> <p>ttymonはpam(3PAM)を使って、セッション管理を行います。PAM構成ポリシーはttymonで使用するモジュールを明記しています。このポリシーは/etc/pam.confで見ることができます。以下にUNIXセッション管理モジュールを使用するttymonコマンドのエントリの入ったpam.confファイルの抜粋を示します。</p> <pre> ttymon session required /usr/lib/security/pam_unix.so.1ttymon </pre> <p>サービスのエントリがない場合にはotherのサービスのエントリを使用します。</p>
オプション	<p>-g ttymonの特殊な呼び出しは-gオプションを指定して行います。コマンドのこの書式を呼び出せるのは、ポートに正しいボーレートおよび端末設定を設定してからloginサービスに接続する必要があるアプリケーションだけにする必要があります。SACの制御下では前もって構成することはできません。-gは、以下のようなオプションの組み合わせとともに使用できます。</p> <p>-d <i>device</i> <i>device</i>はttymonの接続先とすべきポートの完全パス名です。このオプションが指定されていない場合は、ファイル記述子0はTTYポートに対する呼び出しプロセスによって設定される必要があります。</p> <p>-h -hフラグが指定されていないと、ttymonは、速度をデフォルト速度または指定速度に設定する前に、速度を0に設定することによって回線のハングアップを強制的に行います。</p>

ttymon(1M)

- l *ttylabel* *ttylabel* は *ttydefs* ファイルの速度および TTY 定義に対するリンクです。この定義によって、初期の実行速度、初期の TTY 設定の内容およびユーザーが BREAK キーを押して速度が適切でないことを指示する場合に、次に試行する速度が *ttymon* に通知されます。デフォルト速度は 9600 ボーです。
- m *modules* ポートを初期化すると、*ttymon* はポートのモジュールすべてをポップして、指定した順序で *modules* をプッシュします。*modules* はコンマで区切ったプッシュ可能なモジュールのリストです。通常、ポートのデフォルトのモジュールは、オートプッシュファシリティによって設定されます。
- p *prompt* ユーザーはこれを用いるとプロンプト文字列を指定できます。デフォルトのプロンプトは `Login:` です。
- t *timeout* プロンプトの送信後、*timeout* 秒内に何も入力がない場合には *ttymon* を終了します。
- T *termtype* *termtype* に TERM 環境変数を設定します。

環境

LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (*environ*(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応する ロケール のカテゴリにおける *ttymon* の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が *ttymon* の動作を決定します。

LC_CTYPE *ttymon* の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、*ttymon* は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。*ttymon* は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、*ttymon* は 1、2、またはそれ以上のカラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

ファイル

/etc/logindevperm

属性

次の属性については *attributes*(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWcsu

関連項目

ct(1C), *cu*(1C), *autopush*(1M), *pmadm*(1M), *sac*(1M), *sacadm*(1M), *sttydefs*(1M), *ttyadm*(1M), *uucico*(1M), *pam*(3PAM), *logindevperm*(4), *pam.conf*(4), *utmpx*(4), *attributes*(5), *environ*(5), *pam_unix*(5)

Solaris のシステム管理 (第 1 巻)

使用上の留意点

ポートが複数の ttymon によって監視されている場合は、ttymon は入力を争うような方法でプロンプトメッセージを送信できます。

utmp2wtmp(1M)

名前	acct, acctdisk, acctdusg, accton, acctwtmp, closewtmp, utmp2wtmp – アカウンティングおよびいろいろなアカウントコマンドの概要
形式	<pre>/usr/lib/acct/acctdisk /usr/lib/acct/acctdusg [-u filename] [-p filename] /usr/lib/acct/accton [filename] /usr/lib/acct/acctwtmp reason filename /usr/lib/acct/closewtmp /usr/lib/acct/utmp2wtmp</pre>
機能説明	<p>アカウンティングソフトウェアは、アカウンティングシステムを構築するためのツール群です (C 言語のプログラムおよびシェルプロシージャから構成される)。acctsh(1M) では、C 言語のプログラムの一番上に組むシェルプロシージャ群を説明します。</p> <p>接続時間のアカウンティングは記録を /var/adm/wtmpx (utmpx(4) を参照) に書き込むプログラム群により管理されます。acctcon(1M) では、このファイルをセッションおよび課金記録に変換するプログラムを説明します。また、acctmerg(1M) では、この課金記録について集約します。</p> <p>プロセスアカウンティングは、システムカーネルによって行われます。プロセスを終了すると、1つのプロセスにつき1つのレコードが、あるファイル (通常は、/var/adm/pacct) に書き込まれます。acctprc(1M) では、課金のためにこのデータを集約します。コマンドの使用状況を集約するには、acctcms(1M) を使用します。acctcom(1) を用いて、現在のプロセスデータを調査することができます。acctmerg (acct(3HEAD) の tacct フォーマットを参照) を使うと、プロセスのアカウンティングレコードおよび接続時間のアカウンティングレコード (あるいは acct(3HEAD) で説明する tacct フォーマットのアカウンティングレコードのいずれか) をアカウンティングレコードの合計にマージしたり、集約したりすることができます。どのアカウンティングレコードにも、あるいはすべてのアカウンティングレコードにでも prtacct (acctsh(1M) を参照) をフォーマットに使用します。</p> <p>acctdisk は、ユーザー ID、ログイン名およびディスクブロック数がある行を読み取り、他のアカウンティングレコードとマージすることができるアカウンティングレコードの合計に、それらの情報を変換します。入力ファイルが壊れているか、または正しくフォーマットされていない場合は、acctdisk はエラーを返します。</p> <p>acctdusg は、その標準入力を (通常は、find / -print から) 読み取り、ログインによるディスクの資源消費 (間接ブロックを含む) を計算します。</p> <p>accton では、引数がない場合、プロセスアカウンティングをオフにします。filename を指定する場合は、そのファイルは、カーネルがプロセスのアカウンティングレコード (acct(2) および acct(3HEAD) を参照) を追加するための、既存のファイル名である必要があります。</p>

utmp2wtmp(1M)

acctwtmp は、utmpx(4) のレコードを *filename* へ書き込みます。レコードには、現在の時間および *reason* を説明する文字列が入っています。ACCOUNTING のレコードの型が割り当てられます (utmpx(4) を参照)。*reason* には、11 文字以下の文字列、数、\$、または空白を指定してください。たとえば、以下は、レポートのプロシージャおよびシャットダウンのプロシージャでの使用例です。

```
acctwtmp "acctg on" /var/adm/wtmpx
acctwtmp "acctg off" /var/adm/wtmpx
```

現在ログオンしている各ユーザーについて、closewtmp は、偽の DEAD_PROCESS レコードを /var/adm/wtmpx ファイルに入れます。runacct (runacct(1M) を参照) がこの偽の DEAD_PROCESS を使用することにより、接続アカウントングプロシージャは runacct を起動する前にログオンしたユーザーが使用した時間を追跡することができます。

現在ログオンしている各ユーザーについて、runacct は utmp2wtmp を使用して、runacct が作成したファイル /var/adm/wtmpx にエントリを作成します。この /var/adm/wtmpx のエントリを見れば、引き続き起こる runacct の呼び出しで、現在ログインしているユーザーの接続時間がわかります。

オプション 以下のオプションを指定できます。

-u *filename* 誰にも課金されないファイル名からなるレコードを、*filename* に入れます (ディスクの課金を拒否しようとしたユーザーを見つけるための潜在的な情報源となる)。

-p *filename* -p は、パスワードファイル *filename* を指定します。パスワードファイルが /etc/passwd である場合は、このオプションは不要です。

環境 LC_* 変数 (LC_CTYPE、LC_MESSAGES、LC_TIME、LC_COLLATE、LC_NUMERIC、LC_MONETARY) (environ(5) 参照) のいずれも環境に設定されていなければ、それぞれ対応するロケールのカテゴリにおける acct の動作は、環境変数 LANG によって決定されます。もし、LC_ALL が設定されていれば、その内容が LANG 変数やその他の LC_* 変数より優先されます。上記の変数が環境にまったく設定されていなければ、C ロケール (米国スタイル) が acct の動作を決定します。

LC_CTYPE acct の文字の処理方法を決定します。LC_CTYPE に有効な値が設定されていると、acct は、そのロケールにあった文字を含むテキストやファイル名を表示および処理できます。acct は拡張 UNIX コード (EUC) も表示および処理できます。この場合、文字は 1 バイト幅、2 バイト幅、3 バイト幅のいずれも使用できます。また、acct は 1、2、またはそれ以上のカラム幅の EUC 文字も処理することができます。C ロケールにおいては、ISO 8859-1 の文字だけが有効です。

LC_TIME acct の日付および時間のフォーマットの処理方法を決定します。C ロケールにおいては、日付および時間の処理方法は米国ルールに従います。

utmp2wtmp(1M)

ファイル	/etc/passwd	ログイン名からユーザー ID への変換に用いる
	/usr/lib/acct	本マニュアルの 1M 章に含まれるアカウントリングコマンドが置かれる
	/var/adm/pacct	現在のプロセスアカウントリングファイル
	/var/adm/wtmpx	ユーザーアクセスまたは管理情報の履歴

属性 次の属性については attributes(5) のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWaccu

関連項目 acctcms(1M), acctcom(1), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(3HEAD), passwd(4), utmpx(4), attributes(5), environ(5)

Solaris のシステム管理 (第 1 巻)

名前	wall - すべてのユーザーへの通知
形式	<code>/usr/sbin/wall [-a] [-g grpname] [filename]</code>
機能説明	<p>wall はファイルの終わりまで標準入力を読み取り、現在ログインしているすべてのユーザーに以下の文字で始まるメッセージを送信します。</p> <pre>Broadcast Message from ...</pre> <p><i>filename</i> が指定されていれば、メッセージはこのファイルから読み込まれます。通常、リモートログインセッションに対応していない仮想端末は無視されます。したがって、ウィンドウシステムを使用しているときは、メッセージはコンソールウィンドウにだけに表示されます。ただし、<code>-a</code> オプションを指定すると、メッセージを仮想端末などにも送信します。</p> <p>一般的には、システムをシャットダウンする前に、すべてのユーザーに警告を与える場合に、このメッセージを用います。</p> <p>ユーザーが起動させた保護を無効にする場合、送信側はスーパーユーザーである必要があります (<code>mesg(1)</code> 参照)。</p> <p>wall は、他のユーザーの端末上で書き込み権を持つために、グループ ID <code>tty</code> に対して <code>setgid()</code> を実行します (<code>setuid(2)</code> 参照)。</p> <p>wall は、ユーザーの端末に送信する前に非表示可能文字を検出します。制御文字は、適切な ASCII 文字が後に続く <code>^</code> として表示されます。すなわち、高位ビットが設定された文字は <code>meta</code> 表記法で表示されます。たとえば、<code>\003</code> は <code>^c</code>、また <code>\372</code> は <code>M-z</code> と表示されます。</p>
オプション	<p>以下のオプションを指定できます。</p> <p><code>-a</code> コンソールと仮想端末にメッセージを送ります。</p> <p><code>-g grpname</code> <i>grpname</i> で指定されたグループに対してのみメッセージを送ります。</p>
環境	<p><code>LC_*</code> 変数 (<code>LC_CTYPE</code>、<code>LC_TIME</code>、<code>LC_COLLATE</code>、<code>LC_NUMERIC</code>、<code>LC_MONETARY</code>) が環境に設定されていない場合は、それぞれ対応するロケールのカテゴリにおける wall の動作は、環境変数 <code>LANG</code> によって決定されます (<code>environ(5)</code> 参照)。もし、<code>LC_ALL</code> が設定されていれば、その内容が <code>LANG</code> 変数やその他の <code>LC_*</code> 変数より優先されます。上記の変数が環境にまったく設定されていない場合は、<code>C</code> ロケール (米国スタイル) が wall の動作を決定します。</p>
ファイル	<code>/dev/tty*</code>
属性	次の属性については <code>attributes(5)</code> のマニュアルページを参照してください。

属性タイプ	属性値
使用条件	SUNWcsu

wall(1M)

関連項目	mesg(1), write(1), setuid(2), attributes(5), environ(5)
注意事項	wall は、ユーザーの tty ファイルでオープンが失敗すると、Cannot send to . . . と表示します。