



Sun™ ハードウェア Solaris™ リファレンスマニュアル

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303-4900
U.S.A.650-960-1300

Part No. 816-0315-10
Revision A, 2001 年 5 月

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303-4900 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 Limited** 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。本製品のフォント技術を含む第三者のソフトウェアは、著作権法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.

本製品は、株式会社モリサワからライセンス供与されたリュウミン L-KL (Ryumin-Light) および中ゴシック BBB (GothicBBB-Medium) のフォント・データを含んでいます。

本製品に含まれる **HG 明朝 L** と **HG ゴシック B** は、株式会社リコーがリョ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。平成明朝体 W3 は、株式会社リコーが財団法人日本規格協会 文字フォント開発・普及センター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。また、**HG 明朝 L** と **HG ゴシック B** の補助漢字部分は、平成明朝体 W3 の補助漢字を使用しています。なお、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, AnswerBook2, docs.sun.com, Sun StorEdge, Sun Enterprise Network Array 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・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ーに基づくものです。

Java およびその他の **Java** を含む商標は、米国 **Sun Microsystems** 社の商標であり、同社の **Java** ブランドの技術を使用した製品を指します。

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。

ATOK は、株式会社ジャストシステムの登録商標です。**ATOK8** は、株式会社ジャストシステムの著作物であり、**ATOK8** にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。**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。

Netscape, Navigator は、米国 **Netscape Communications Corporation** の商標です。**Netscape Communicator** については、以下をご覧ください。Copyright 1995 Netscape Communications Corporation. All rights reserved.

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** 社が自社のユーザー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・ユーザーインター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。

本書には、技術的な誤りまたは誤植のある可能性があります。また、本書に記載された情報には、定期的に変更が行われ、かかる変更は本書の最新版に反映されます。さらに、米国サンまたは日本サンは、本書に記載された製品またはプログラムを、予告なく改良または変更することがあります。

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： *Solaris on Sun Hardware Reference Manual Supplement*
Part No: 816-0239-10
Revision A



Please
Recycle



Adobe PostScript

目次

はじめに v

1. ユーザーコマンド

cdrw(1) 2

2. 保守コマンド

cfgadm_sbd(1M) 10

envmond(1M) 27

nf_fddidaemon(1M) 29

nf_install_agents(1M) 30

nf_macid(1M) 31

nf_smtmon(1M) 32

nf_snmd(1M) 34

nf_snmd_kill(1M) 36

nf_stat(1M) 37

nf_sync(1M) 41

pf_fddidaemon(1M) 42

pf_install_agents(1M) 43

pf_macid(1M) 44

pf_smtmon(1M) 45
pf_snmd(1M) 47
pf_snmd_kill(1M) 49
pf_stat(1M) 50
rscadm(1M) 54
sunvts(1M) 59
vts_cmd(1M) 61
vtsk(1M) 68
vtsprobe(1M) 70
vtstty(1M) 73
vtsui(1M) 76
vtsui.ol(1M) 77

3. ファイル形式

envmond.conf(4) 80

4. デバイスおよびネットワークインタフェース

nf(7) 84
pf(7) 88
smt(7) 95

5. デバイス

ge(7D) 100

はじめに

このマニュアルには、サンのハードウェアを使用しているユーザーに対して提供される **Solaris 8** ソフトウェア製品のリファレンスマニュアル (マニュアルページ) が含まれています。このマニュアルは、『**Solaris 8** リファレンスマニュアル』に記載されているマニュアルページを補足するマニュアルです。この版には、**Solaris 8 4/01** リリースで改訂されたマニュアルページの内容が含まれています。

man コマンドを使用してこのマニュアルに記載されている情報を参照するには、事前に、当該 **Solaris** リリースに対応する **Sun Microsystems Computer Systems Supplement CD** からソフトウェアをインストールする必要があります。通常、**Sun Microsystems Computer Systems Supplement CD** からソフトウェア製品をインストールすると、当該ソフトウェアのマニュアルページが含まれているパッケージも自動的にインストールされます。マニュアルページのインストール方法の詳細については『**Solaris 8 Sun** ハードウェアマニュアル』を参照してください。

このマニュアルの構成

このマニュアルでは、マニュアルページを下記のカテゴリごとにアルファベット順に記載しています。

- ユーザーコマンド (1)
- 保守コマンド (1M)
- ファイル形式 (4)
- デバイスおよびネットワークインタフェース (7)
- デバイス (7D)

下記の製品のマニュアルページが含まれています。

- CD 読み書き用ドライバ: cdrw

- SunFDDI™ ネットワークアダプタソフトウェア : `nf`, `nf_fddidaemon`,
`nf_install_agents`, `nf_macid`, `nf_smtmon`, `nf_snmd`, `nf_snmd_kill`,
`nf_stat`, `nf_sync`, `pf`, `pf_fddidaemon`, `pf_install_agents`, `pf_macid`,
`pf_smtmon`, `pf_snmd`, `pf_snmd_kill`, `pf_stat`, `smt`
- Sun Remote System Control (RSC): `rscadm`
- SunVTS™ 診断ソフトウェア : `sunvts`, `vts_cmd`, `vtsk`, `vtsprobe`, `vtstty`,
`vtsui`, `vtsui.ol`
- Netra™ ct サーバー用環境モニターソフトウェア : `envmond`, `envmond.conf`

書体と記号について

このマニュアルで使用している書体と記号について説明します。

表 P-1 このマニュアルで使用している書体と記号

書体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上的コンピュータ出力、コード例。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を実行します。 % You have mail.
AaBbCc123	ユーザーが入力する文字を、画面上的コンピュータ出力と区別して表します。	% su Password:
<i>AaBbCc123</i> またはゴシック	コマンド行の可変部分。実際の名前や値と置き換えてください。	<code>rm filename</code> と入力します。 <code>rm ファイル名</code> と入力します。
『 』	参照する書名を示します。	『Solaris ユーザーマニュアル』
「 」	参照する章、節、または、強調する語を示します。	第 6 章「データの管理」を参照。 この操作ができるのは「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅をこえる場合に、継続を示します。	% <code>grep `^#define \ XV_VERSION_STRING`</code>

シェルプロンプトについて

シェルプロンプトの例を以下に示します。

表 P-2 シェルプロンプト

シェル	プロンプト
UNIX の C シェル	マシン名%
UNIX の Bourne シェルと Korn シェル	\$
スーパーユーザー (シェルの種類を問わない)	#

第1章

ユーザーコマンド

名前	<code>cdrw</code> — CD 読み書きコマンド
形式	<pre> cdrw -i [-hvSCO] [-d デバイス] [-p 書き込み速度] [イメージファイル] cdrw -a [-hvSCO] [-d デバイス] [-p 書き込み速度] [-T 音楽形式] 音楽ファイル1 音楽ファイル2... cdrw -x [-hv] [-d デバイス] [-T 音楽形式] トラック番号 出力ファイル cdrw -c [-hvSC] [-d デバイス] [-p 書き込み速度] [-m 一時ディレクトリ] [-s ソースデバイス] cdrw -b [-hv] [-d デバイス] all session cdrw -M [-hv] [-d デバイス] cdrw -l [-hv] </pre>
使用条件	SUNWcdrw
機能説明	<p><code>cdrw</code> コマンドを用いてデータ CD および音楽 CD を作成することができます。また音楽 CD から音楽トラックを取り出すことができます。MMC 準拠の CD-R/CD-RW ドライブであれば、どれでも <code>cdrw</code> で使用することができます。</p> <p><code>-d</code> オプションによりデバイスを指定しない場合は、<code>cdrw</code> はシステムに接続された CD ライターを探します。システム上に 1 つのライターデバイスを発見したときは、それをデフォルト CD ライターとして扱います。</p> <p>2 つ以上の CD ライターがシステムに接続されている場合は、<code>-d</code> オプションによりデバイスを指定してください。CD ライターのデバイス名は以下のように指定されます。/dev/rdisk/cTnTdNsN, cTnTdNsN, cTnTdN, また、ボリュームマネージャで使用する <code>cdrom</code>, <code>cdrom1</code> などのシンボル名でも指定できます。<code>-l</code> オプションによって CD ライターのリストを表示します。</p>
データ CD の作成	<p>データ CD を作成する場合 <code>cdrw</code> はトラック書き込みモード (track-at-once) を使用します。<code>-i</code> オプションを用いて CD メディアに書き込むデータのファイルを指定します。ファイルが指定されない場合は <code>cdrw</code> は標準入力からデータを読み込みます。</p>

音楽 CD の作成

どちらの場合でも、データはあらかじめ `mkisofs(1M)` コマンドを使用してファイルおよびファイル情報を **CD** で使用される **High Sierra** フォーマットに変換しておきます。コマンドの使用方法は「使用例」を参照してください。

音楽 **CD** を作成する場合、`-a` オプションを用いて 1 つもしくは複数の音楽ファイルを指定します。すべての音楽ファイルはサポートされた音楽形式である必要があります。現在対応している形式は以下のものです。

<code>sun</code>	レッドブック CD-DA 形式の Sun .au ファイル
<code>wav</code>	レッドブック CD-DA 形式の RIFF (.wav) ファイル
<code>cda</code>	生の CD 音楽データを含む .cda ファイル (リトルエンディアン 16 bit PCM ステレオ 44.1 KHz サンプリング)
<code>aur</code>	生の CD データを含む .aur ファイル (ビッグエンディアン)

音楽形式が指定されない場合は `cdrw` はファイルの拡張子により音楽形式を認識しようとします。拡張子の太文字小文字は無視されます。`-T` により音楽形式を指定した場合は指定されたすべてのファイルの音楽形式として扱われます。また `cdrw` は音楽トラックを書き込んだ後セッションをクローズします。したがって、書き込まれるトラックは **1** つのコマンド行で指定する必要があります。

音楽の取り出し

`cdrw` は `-x` オプションを用いて音楽 **CD** から音楽データを取り出すことにも使用できます。**CD** にはレッドブック **CD-DA** 形式のトラックが存在する必要があります。デフォルトの出力形式はファイルの拡張子に依存します。ユーザーは `-T` オプションにより出力形式を `sun, wav, cda, au, aur` に指定することができます。

CD のコピー

`cdrw` はシングルセッションのデータ **CD-ROM** やレッドブックの音楽 **CD** のコピーに使用することができます。**CD** をコピーする場合 `cdrw` は指定されたソースデバイスを探します。`-c` オプションを使用時にソースデバイスが指定されない場合は、現在の書き込み用 **CD** デバイスがソースデバイスとみなされます。`cdrw` は一時ファイルに音楽トラックを取り出し、現在の書き込み用 **CD** デバイ스에 ブランクの書き込み可能 **CD-R/CD-RW** メディアが入っているか調べます。メディアが見つからない場合は、現在の書き込み用 **CD** デバイ스에 ブランクの書き込み可能 **CD** メディアを入れるようユーザーに要求します。デフォルトの一時ディレクトリに十分な領域がない場合は `-m` オプションを用いて代替のディレクトリを指定します。

CD-RW メディアの消去

ユーザーは再書き込みする前に **CD-RW** メディアを消去する必要があります。
 -b オプションは以下の消去方法をサポートしています。

`session` 最後のセッションを消去する
`all` メディア全体を消去する。

`session` 消去では `cdrw` は最後のセッションを消去します。セッションが 1 つしか書き込まれていない **CD-RW** (`cdrw` によって作成されたデータ/音楽 **CD-RW**) の場合は、記録された部分を消去するだけで空白ディスクにすることができます。これはメディア全体を消去するよりも速くできます。

`all` 消去は、マルチセッションの **CD**、最後のセッションがクローズされていない場合、**CD** の状態が不明な場合、ユーザーが **CD** 全体を消去したい場合に使用してください。この場合 `cdrw` はディスク全体を消去します。

-l オプションによって現在システムに接続されている **CD** ライターの一覧が表示されます。また、メディアによっては -M オプションによって、消去の状態や内容リスト (**TOC**) が表示されます。また -M オプションは、最後のセッションの開始アドレスや次の書き込み可能アドレスも表示します。この情報は -o オプションを指定してマルチセッション **CD** を作成するときと一緒に用いられます。詳細は `mkisofs(1M)` を参照してください。

デバイスリスト
および
メディアの状態の
チェック

オプション

- a 音楽 **CD** を作成します。少なくとも 1 つの音楽ファイルを指定する必要があります。最大 99 までの音楽トラックを指定することができます。また最長の音楽データは -C オプションを指定しない場合デフォルトで 74 分となります。
- b **CD-RW** メディアを消去します。消去する方法は `all` もしくは `session` のどちらかを指定します。
- c **CD** をコピーします。他に引き数の指定がない場合は、デフォルトの **CD** 書き込みデバイスをソースデバイスとみなします。その場合、コピー処理はソースメディアから一時ディレクトリに読み出し、ユーザーに空白メディアをドライブに入れるよう要求します。
- c メディアの容量を指定します。このオプションがない場合 `cdrw` で書き込み可能な **CD** メディアの容量のデフォルト値は、音楽 **CD** では 74 分、データ **CD** では 681984000 (約 650M) バイトとなります。

- d **CD** 書き込みを行うデバイスを指定します。
- h ヘルプ。使用法を表示します。
- i データ **CD** を作成するイメージファイルを指定します。書き込めるファイルサイズは **CD-R/CD-RW** メディアのいずれの場合でもデフォルトの **681984000** バイト、もしくは **-c** オプションによって指定された容量以下となります。**CD** 書き込み処理は、途切れない連続したデータの供給を必要とするため、イメージファイルは **NFS** マウントされたファイルシステムよりもローカルなファイルシステムに置くようにしてください。
- l システム上のすべての **CD** ライターを表示します。
- m システムのデフォルトの一時ディレクトリに代わって **CD** をコピーする際にトラックデータを置く一時ディレクトリを指定します。代わりの一時ディレクトリには十分な領域が必要です。**(80 分の音楽 CD の場合で約 800MB)** システムのデフォルト一時ディレクトリは十分な容量を持たないことがあります。
- M メディアの状態を表示します。cdrw はメディアがブランクかどうか、内容リスト (**TOC**)、最後のセッションの開始アドレス、ディスクがオープンな場合は次の書き込み可能アドレス、を表示します。
- O ディスクをオープンのままにします。cdrw は通常セッションをクローズしますが、マルチセッション **CD** を作成する場合は次のセッションを追加するために、オープンのままにしておきます。
- p **CD** ライターの書き込み速度を設定します。たとえば **-p 4** は **4** 倍速で書き込みます。オプションの指定がない場合は cdrw は **CD** ライターのデフォルトの書き込み速度を使用します。オプションが指定された場合は cdrw はドライブに指定された速度での書き込みを設定しようとしませんが実際にドライブがその速度で書き込めるか保証はありません。
- s **CD** をコピーするソースデバイスを指定します。
- S 疑似書き込みモードです。このモードでは cdrw は **CD** ライターのレーザーをオフにして書き込みを行います。したがってメディアにはデータは書き込まれません。システムが **CD** ライターに指定された書き込み速度で、十分にデータを提供できるかどうか調べる場合に使用します。

使用例

- T 音楽 CD を作成するために読み出す音楽ファイルもしくは取り出す音楽ファイルの形式を指定します。音楽形式は sun, wav, cda, au, aur が可能です。
- v 冗長モードです。
- x 音楽トラックから音楽データを取り出します。

例 1 データ CD の作成。

```
cdrw -i /local/iso_image
```

例 2 /home/foo ディレクトリ以下を CD に書き込む。

```
$ mkisofs -r /home/foo 2>/dev/null | cdrw -i -p 1
```

例 3 1 番目の音楽トラックを /home/foo/song1.wav に取り出す。

```
$ cdrw -x -T wav 1 /home/foo/song1.wav
```

例 4 ディスク上の wav ファイルから音楽 CD を作成する。

```
$ cdrw -a song1.wav song2.wav song3.wav song4.wav
```

例 5 CD-RW ドライブ内の CD-RW メディアのデータを消去する。

```
$ cdrw -b all
```

例 6 複数の CD-R/RW ドライブが接続されたシステムでデータ CD を作成する。

```
$ cdrw -d c1t6d0s2 -i /home/foo/iso-image
```

例 7 システムが書き込み処理に対して十分な速度でデータを CD-RW ドライブに提供できるかどうかチェックする。

```
$ cdrw -S -i /home/foo/iso-image
```

例 8 cdrw を優先度を上げて実行する (root ユーザーのみ)

```
# priocntl -e -p 60 cdrw -i /home/foo/iso-image
```

関連項目

mkisofs(1M), audioconvert(1)

注意事項

CD 書き込み中、システムは一定の転送速度でドライブにデータを供給し続ける必要があります。CD 作成中は I/O 処理を最小限にし、不要なアプリケーションは終了しておくことを推奨します。

コピーの作成や音楽トラックの取り出しには MMC 準拠のソース CD-ROM ライブの使用をお勧めします。CD ライターはこの目的のために使用されます。

CD に書き込む前に、メディアがブランクかどうか `-M` オプションで確認し、システムが要求された書き込み速度でデータを供給できるかどうか `-s` オプションで調べてください。システムが指定された書き込み速度でデータを供給できない場合は `-p` オプションにより、書き込み速度を遅くしてください。また `priocntl(1)` を用いて優先度を高くして `cdrw` を実行することもできます。

`-p` オプションは **CD-R/RW** ドライブに詳しく、書き込み速度を変更して使用できるユーザーのために提供されています。一般に普及しているドライブでは、ドライブによって書き込み速度の設定コマンドの扱いが異なるため、注意してこのオプションを使用してください。

一般に普及しているドライブでは 74 分を越えて (80 分メディア等) メディアの容量の上限まで書き込むことができます。しかし、使用しているドライブでは 74 分以上の書き込みをサポートしていないかもしれません。使用しているドライブが 74 分以上書き込める場合は、`-c` オプションを使用してメディアの容量を指定するようにしてください。

注 -市販の音楽 CD 等を作者の許諾なく複製することは、個人的に楽しむなどの他は、著作権法上禁じられています。

cdrw(1)

ユーザーコマンド

第2章

保守コマンド

名前	cfgadm_sbd — システムボード管理用 cfgadm コマンド										
形式	<pre> cfgadm -l [-a] [-o parsable] ap_id... cfgadm -c <i>function</i> [-f] [-y -n] [-o unassign nopoweroff] ap_id... cfgadm -t [-f] ap_id... cfgadm -x <i>function</i> ap_id... </pre>										
機能説明	<p>sbd.so.1 プラグインは、sbd クラスのシステムボードの接続、構成、構成解除、および切り離しを行う際の動的再構成機能を提供します。また、システムをリブートすることなく、システムの動作中にシステムボードを接続、切り離すことも可能にします。</p> <p>cfgadm コマンドは、<code>/usr/sbin</code> に置かれています。cfgadm(1M) を参照してください。</p> <p>各ボードスロットは、デバイスツリーで、単一の接続点 (attachment point) として表示されます。各構成要素は、動的接続点として表示されます。各構成要素のタイプ、状態 (state)、条件 (condition)、および各ボードスロットの状態と条件は、<code>-a</code> オプションを使って表示できます。</p> <p>[構成要素タイプ]</p> <p>次は、構成要素タイプの名前と機能説明です。</p> <table data-bbox="570 1247 922 1398"> <tr> <td>cpu</td> <td>CPU</td> </tr> <tr> <td>pci</td> <td>I/O デバイス</td> </tr> <tr> <td>memory</td> <td>メモリー</td> </tr> </table> <p>注意: あるメモリー構成要素上の操作は、ボード上のすべてのメモリー構成要素に影響します。</p> <p>[構成要素の条件]</p> <p>次は、構成要素条件の種類と機能説明です。</p> <table data-bbox="570 1650 1170 1745"> <tr> <td>failed</td> <td>構成要素はテストに失敗しました。</td> </tr> <tr> <td>ok</td> <td>構成要素は機能しています。</td> </tr> </table>	cpu	CPU	pci	I/O デバイス	memory	メモリー	failed	構成要素はテストに失敗しました。	ok	構成要素は機能しています。
cpu	CPU										
pci	I/O デバイス										
memory	メモリー										
failed	構成要素はテストに失敗しました。										
ok	構成要素は機能しています。										

unknown 構成要素はテストされていません。

[構成要素の状態]

次は、構成要素の受容体 (**receptacle**) 状態の名前と機能説明です。

connected 構成要素がボードスロットに接続されています。

次は、構成要素の占有装置 (**occupant**) 状態の名前と機能説明です。

configured 構成要素は、**Solaris** オペレーティング環境で使用可能です。

unconfigured

構成要素は、**Solaris** オペレーティング環境で使用可能な状態にありません。

[ボードの条件]

次は、ボード条件の種類と機能説明です。

failed ボードはテストに失敗しました。

ok ボードは機能しています。

unknown ボードはテストされていません。

unusable ボードスロットは使用できません。

[ボードの状態]

ボードを取り付けると、受容体状態は、**empty** から **disconnected** に変わります。ボードを取り外すと、受容体状態は、**disconnected** から **empty** に変わります。

警告: **connected** 状態にあるボード、または電源が投入され、**disconnected** 状態にあるボードを取り外すと、オペレーティングシステムに障害が発生し、システムが回復不能な損傷を受ける場合があります。

次は、ボードの受容体状態の名前と機能説明です。

connected ボードは、電源が投入され、システムバスに接続されています。ボード上の構成要素を表示できるのは、ボードが **connected** 状態になった後に限ります。

disconnected

ボードは、システムバスから切り離されています。**disconnected** 状態でも、ボードには、電源が投入されている場合があります。したがって、ボードの電源を切断し、**disconnected** 状態にしてから、ボードをスロットから取り外してください。

empty ボードが存在しません。

disconnected 状態のボードの占有装置状態は、常に、**unconfigured** です。次の表は、ボードの占有装置状態の名前と機能説明です。

configured 少なくとも、ボード上の 1 つの構成要素が構成されています。

unconfigured

ボード上のすべての構成要素が構成解除されています。

[動的システムドメイン (DSD: Dynamic System Domain)]

動的システムドメイン (**DSD**、このマニュアルではドメインという) に基づくプラットフォームは、シャーシのスロットを電気的に分離されたハードウェアパーティション (すなわち、**DSD**) に分割します。**DSD** に基づいていないプラットフォームは、永久的にすべてのスロットをシステムに割り当てます。

スロットは、空でも実装されていてもかまいません。また、任意の数のドメインを使用可能にすることも、割り当てることもできます。あるドメインが使用できるスロットの数は、システムコンローラ上で維持されている使用可能構成要素リスト (**ACL: Available Component List**) により制御されます。**ACL** は、**Solaris** オペレーティング環境におけるアクセス制御リストではありません。

スロットは、あるドメインの **ACL** にあり、別のドメインに割り当てられていない場合に限り、その特定のドメインで認識可能になります。未割り当てスロットは、**ACL** 内のスロットをもっているドメインすべてにおいて認識可能になります。スロットがあるドメインに割り当てられると、そのスロットは、他のドメインでは認識できなくなります。

あるドメインで認識可能であり、未割り当てのスロットは、まず、そのドメインに割り当てをしてから、別の状態変更コマンドを適用する必要があります。この割り当ては、**-x assign** を使って明示的に行うことも、暗黙的に **connect** の一部として行うこともできます。スロットは、別のドメインが使用するためには、まず、割り当てを解除する必要があります。割り当て解除は、**-x unassign** を直接使用するか、または **disconnect** のオプションとして **-o unassign** を使って、常に明示的に行います。

[状態変更関数]

ボードスロットの状態またはボード上の構成要素の状態を変更する関数は、どの接続点に対しても同時に実行できます。状態変更操作は、一度に一回しか許されません。状態変更情報の **Busy** フィールドの **Y** は、操作を実行中であることを示します。

状態変更関数を次に示します。

- **configure**
- **unconfigure**
- **connect**
- **disconnect**

[可用性変更関数]

ボードの可用性を変更するコマンドは、どの接続点に対しても同時に実行できます。可用性変更操作は、一度に一回しか許されません。この関数は、**cfgadm -l** 出力中の情報文字列も変更します。**Busy** フィールドの **Y** は、操作を実行中であることを示します。

可用性変更関数を次に示します。

- assign
- unassign

[条件変更関数]

ボードスロットまたはボード上の構成要素の条件を変更する関数はどの接続点に対しても同時に実行できます。条件変更操作は、一度に一回しか許されません。この関数は、`cfgadm -l` 出力中の情報文字列も変更します。**Busy** フィールドの **Y** は、操作を実行中であることを示します。

条件変更関数を次に示します。

- poweron
- poweroff
- test

[構成解除プロセス]

ここでは、構成解除プロセスについて説明します。特に、構成解除プロセスの異なる段階での、コピーと名前変更時の操作元と操作先のボード状態を具体的に示します。

次のコーディング例では、ボード **0** 上の永久メモリーをドメインの別のボードに移動しておく必要があります。したがって、ボード **0** が操作元、ボード **1** が操作先となります。

状態変更操作は、ビジーとなっているボードでは、起動できません。説明を簡単にするため、**CPU** 情報は、コーディング例から省略してあります。

構成解除プロセスは、次のコマンドで開始します。

```
# cfgadm -c unconfigure -y sbd/slot0::memory &
```

まず、ボード **0** の永久メモリーと同じアドレス範囲にあるボード **1** 上のメモリーを削除する必要があります。このフェーズの間、操作元ボード、操作先ボード、およびメモリー接続点は、ビジーとなります。状態は、次のコマンドを使って表示できます。

```
# cfgadm -a -s cols=ap_id:type:r_state:o_state:busy \
sbd/slot0 sbd/slot1
```

Ap_Id	Type	Receptacle	Occupant	Busy
sbd/slot0	cpu/mem	connected	configured	y
sbd/slot0::memory	memory	connected	configured	y
sbd/slot1	cpu/mem	connected	configured	y
sbd/slot1::memory	memory	connected	configured	y

ボード 1 上のメモリーの削除が終了すると、ボード 1 は、**unconfigured** となります。ボード 0 上のメモリーは、次の例のように、**configured** のままですが、やはり、ビジーとなります。

Ap_Id	Type	Receptacle	Occupant	Busy
sbd/slot0	cpu/mem	connected	configured	y
sbd/slot0::memory	memory	connected	configured	y
sbd/slot1	cpu/mem	connected	configured	y
sbd/slot1::memory	memory	connected	unconfigured	n

その後、ボード 0 のメモリーが、ボード 1 にコピーされます。コピーが終了すると、メモリーの占有装置状態が切り替わります。ボード 0 上のメモリーは、**unconfigured** となり、ボード 1 上のメモリーが **configured** となります。プロセスのこの時点で、次の例のように、ボード 0 だけがビジーのままになります。

Ap_Id	Type	Receptacle	Occupant	Busy
sbd/slot0	cpu/mem	connected	configured	y
sbd/slot0::memory	memory	connected	unconfigured	n
sbd/slot1	cpu/mem	connected	configured	n
sbd/slot1::memory	memory	connected	configured	n

プロセス全体が終了した後、次の例のように、ボード 0 上のメモリーは **unconfigured** のままであり、接続点はビジーではありません。

Ap_Id	Type	Receptacle	Occupant	Busy
sbd/slot0	cpu/mem	connected	configured	n
sbd/slot0::memory	memory	connected	unconfigured	n
sbd/slot1	cpu/mem	connected	configured	n
sbd/slot1::memory	memory	connected	configured	n

ページング不可能メモリーは移動され、ボード 0 上のメモリーは **unconfigured** となります。この時点で、どちらのボード上でも、新しい状態変更操作を実行することができます。

オプション

[プラットフォーム固有オプション]

プラットフォーム固有オプションは、システムボードプラグインが解釈するオプションに続いて指定できます。プラットフォーム固有オプションはすべて、`platform` キーワードの後に指定しなければなりません。次の例は、プラットフォーム固有オプションをもつコマンドの一般的な形式です。

```
command -o sbd_options,platform=platform_options
```

このマニュアルページには、`cfgadm` コマンド用の `-v` オプション、`-s` オプション、または `-h` オプションの機能説明はありません。これらのオプションについての機能説明は、`cfgadm(1M)` を参照してください。次のオプションが、`cfgadm_sbd` プラグインによってサポートされています。

-c *function*

状態変更関数を実行します。次の関数を使うことができます。

unconfigure

占有装置状態を **unconfigured** に変更します。この関数は、システムボード上のすべての構成要素およびシステムボードスロットに適用されます。

`unconfigure` 関数は、CPU リストから CPU を削除した後、システムメモリープールから物理メモリーを削除します。使用中のデバイスがある場合は、`cfgadm` コマンドは失敗し、障害をユーザーに報告します。デバイスがビジーでなくなると、すぐにコマンドを再試行できます。CPU が使用中の場合は、コマンドを実行する前に、CPU をオフラインにしておきます。`pbind(1M)`、`psradm(1M)`、および `psrinfo(1M)` を参照してください。

`unconfigure` 関数は、物理メモリーを他のシステムボードに移動した後、構成解除するボードからメモリーを削除します。移動するメモリータイプにより、別のボード上に十分なメモリーを発見できない場合、または該当する物理メモリー範囲を発見できない場合は、コマンドが失敗します。

永久メモリーの場合は、メモリーが移動され、メモリーコントローラが再プログラミングされている間、オペレーティングシステムを停止 (すなわち、休止) しなければなりません。オペレーティングシステムの休止が確認された場合は、構成解除操作に進むように要求されます。yes または no で答えるには、それぞれ、-y オプションまたは -n オプションを常に使ってください。

メモリーの移動は、メモリー容量とシステム負荷により、終了するまで数分かかる場合があります。メモリー移動操作の進行状況は、メモリー接続点に対して、status コマンドを実行して監視できます。また、cfgadm コマンドを停止して、メモリー移動操作を中断することもできます。削除されたメモリーは、システムメモリープールに戻されます。

disconnect

受容体状態を **disconnected** に変更します。この関数は、システムボードスロットだけに適用されます。

占有装置状態が **configured** の場合は、disconnect 関数は、占有装置を構成解除しようとします。構成解除後、システムボードの電源を切断します。この時点で、ボードをスロットから取り外すことができます。

この関数は、動的システムドメインをサポートするプラットフォーム上では、ボードを **assigned** 状態のままにします。

-o nopoweroff を指定すると、disconnect 関数は、ボードを電源投入状態のままにします。-o unassign を指定すると、disconnect 関数は、ドメインからボードを割り当て解除します。

ボードがドメインから割り当て解除されると、別のドメインに割り当てることができます。割り当てられた場合は、割り当てを解除された元のドメインは、ボードの使用ができなくなります。

configure

占有装置状態を **configured** に変更します。この関数は、システムボード上のすべての構成要素とシステムボードスロットに適用されます。

受容体状態が **disconnected** の場合、configure 関数は、受容体を接続しようとしています。接続後、connect 関数によって作成されたデバイスツリーを順に調べていき、必要であれば、デバイスを接続します。この関数を実行すると構成済みの構成要素を除き、ボード上のすべての構成要素が構成されます。

複数の CPU の場合、configure 関数は、複数の CPU を CPU リストに追加します。メモリーの場合、configure 関数は、メモリーを確実に初期化した後、メモリーをシステムメモリープールに追加します。追加された複数の CPU とメモリーは、configure 関数が正常に終了した後、いつでも使用できる状態になります。

I/O デバイスの場合、デバイスを使用可能にするには、まず、mount コマンドと ifconfig コマンドを使用する必要があります。ifconfig(1M) と mount(1M) を参照してください。

connect

受容体状態を **connected** に変更します。この関数は、システムボードスロットだけに適用されます。

ボードスロットがドメインに割り当てられていない場合は、connect 関数は、スロットをドメインに割り当てようとしません。次に、ボードの電源を投入し、テストを行った後、ボードをシステムバスに電氣的に接続し、構成要素を調べます。

connect 関数が正常に終了すると、-a オプションを使ってボード上の構成要素の状態を表示できます。connect 関数は、すべての構成要素を **unconfigured** 状態のままにします。

割り当てステップは、動的システムドメインをサポートするプラットフォームだけに適用されます。

- f ソフトウェアの状態変更の制約を無効にします。-t オプションとともに指定すると、-f オプションは、テスト済みである場合でも、システムボードを強制的にテストします。

-f オプションが、ハードウェアとオペレーティングシステムの根本的な安全性と可用性に対する制約を無効にすることは絶対にありません。

- l 各接続点タイプに応じて、一般的な情報とプラットフォーム固有情報を一覧表示します。プラットフォーム固有情報は、info フィールドに追加されます。

構文解析が可能な info フィールドは、次項から構成されています。

cpu

cpu タイプは、次の情報を表示します。

cpuid=#

ここで、# は CPU の ID を表した番号です。

speed=#

ここで、# は CPU のスピードを MHz 単位で表した数値です。

ecache=#

ここで、# は E-cache のサイズを M バイト単位で表した数値です。

memory

memory タイプは、該当する場合に次の情報を表示します。

address=#

ここで、# はベース物理アドレスを表した数値です。

size=#

ここで、# はメモリーのサイズを **K** バイト単位で表した数値です。

permanent=#

ここで、# はページング不可能メモリーのサイズを **K** バイト単位で表した数値です。

unconfigurable

メモリーを構成解除できないようにするオペレーティングシステムの設定です。

inter-board-interleave

ボードが、他のボードとのインタリーブに参加します。

source=*ap_id*

操作元の接続点を表します。

target=*ap_id*

操作先の接続点を表します。

deleted=#

ここで、# は削除済みのメモリーの容量を **K** バイト単位で表した数値です。

remaining=#

ここで、# はまだ削除できるメモリーの容量を **K** バイト単位で表した数値です。

io

io タイプは、次の情報を表示します。

device=*path*

I/O 構成要素に対する物理パスを表します。

referenced

I/O 構成要素が参照されています。

board

board タイプは、次の情報を表示します。

assigned

ボードがドメインに割り当てられています。

powered-on

ボードの電源が投入されています。

-o parsable を指定しないと、同じ項目がもっと読みやすい形式で、info フィールドに表示されます。

-o parsable

ブール名、または一連の「名前=値」ペアとして info フィールドに情報を戻します。「名前=値」ペアは、1 つのスペース文字で区切れ、二重引用符で囲まれます。二重引用符とバックスラッシュ(¥)文字からは、1 つのバックスラッシュ(¥)で抜けます。ブール名が存在しない場合は、反対が適用されることを示します。

-o parsable オプションは、-s オプションとともに使うことができます。-s オプションについては、**cfgadm(1M)** マニュアルページを参照してください。

-t ボードをテストします。

ボードは、適切なレベルのテストに合格していなければ、接続できません。デフォルトでは、ボードは、適切なレベルのテストに合格しているため、再テストする必要はありません。ただし、-f オプションを使用して、テストをもう一度強制的に実行できます。

-x *function*

sbd-class 関数を実行します。次の関数を使うことができます。

assign

1 つのボードをあるドメインに割り当てます。

受容体状態は、**disconnected** または **empty** である必要があります。ボードは、ドメイン使用可能構成要素リストに載っている必要もあります。動的システムドメインを参照してください。

unassign

あるドメインから 1 つのボードの割り当てを解除します。

受容体状態は、**disconnected** または **empty** である必要があります。ボードは、ドメイン使用可能構成要素リストに載っている必要もあります。動的システムドメインを参照してください。

poweron

システムボードの電源を投入します。

受容体状態は、**disconnected** である必要があります。

poweroff

システムボードの電源を切断します。

受容体状態は、**disconnected** である必要があります。

オペランド

次のオペランドがサポートされています。

Receptacle *ap_id*

受容体接続点 ID は、形式 `sbd/slot_nameX` を取ります。ここで、**X** はスロット番号と等しくなります。

Component *ap_id*

構成要素接続点 ID は、形式 `component_typeX` を取ります。ここで、**component_type** は、「構成要素タイプ」に記載されている構成要素タイプの 1 つと、**X** はスロット番号とそれぞれ等しくなります。

構成要素番号は、ボード相対ユニット番号です。

使用例

例 1: システムボードのすべての接続点を一覧表示

```
# cfgadm -a -s "select=class(sbd)"
Ap_Id          Type      Receptacle  Occupan      Condition
sbd/slot0      cpu/mem  connected   configured   ok
sbd/slot0::    cpu      connected   configured   ok
sbd/slot0::memory memory  connected   configured   ok
sbd/slot1      pci      connected   configured   ok
sbd/slot1::pci0 pci      connected   configured   ok
sbd/slot1::pci1 pci      connected   configured   failed
sbd/slot2      cpu/mem  disconnected unconfigured failed
sbd/slot3      cpu/mem  disconnected unconfigured unknown
sbd/slot4      unknown  empty       unconfigured unusable
```

この例は、次の条件を表形式で表現したものです。

- スロット 1 の 2 番目の PCI ノードはテストに失敗しました。
- スロット 2 のボードはテストに失敗しました。
- スロット 4 は使用不可です。したがって、そのスロットにボードをホットプラグできません。

例 2: システムボードの接続点上のすべての CPU を一覧表示

```
# cfgadm -a -s "select=class(sbd):type(cpu)"
Ap_Id          Type      Receptacle  Occupant      Condition
sbd/slot0::cpu0  cpu      connected   configured   ok
sbd/slot0::cpu1  cpu      connected   configured   ok
sbd/slot0::cpu2  cpu      connected   configured   ok
sbd/slot0::cpu3  cpu      connected   configured   ok
```

例 3: CPU 情報フィールドを表示

```
# cfgadm -l -s noheadings,cols=info sbd/slot0::cpu0
cpuid 16 speed 400 Mhz ecache 8 Mbytes
```

例 4: 構文解析が可能な形式で CPU 情報フィールドを表示

```
# cfgadm -l -s noheadings,cols=info -o parsable \
sbd/slot0::cpu0
"cpuid=16", "speed=400", "ecache=8"
```

例 5: I/O ボード上のデバイスを表示

```
# cfgadm -a -s noheadings,cols=ap_id:info -o parsable \
sbd/slot1
sbd/slot1::pci0 "device=/devices/saf@0/pci@0,2000" referenced
sbd/slot1::pci1 "device=/devices/saf@0/pci@1,2000" referenced
```

例 6: 構成解除操作を監視

この例では、メモリーサイズは M バイト単位で表示されます。

```
# cfgadm -c unconfigure -y sbd/slot0::memory &
# cfgadm -l -s noheadings,cols=info -o parsable \
sbd/slot0::memory sbd/slot1::memory

"address=0x0", "size=16384", "target=sbd/slot1::memory",
"deleted=1240", "remaining=6144",

"address=0x1000000", "size=16384", "source=sbd/slot0::memory"
```

例 7: ドメインにスロットを割り当て

```
# cfgadm -x assign sbd/slot2
```

例 8: ドメインからスロットを割り当て解除

```
# cfgadm -x unassign sbd/slot3
```

ファイル

次のファイルがサポートされています。

/usr/platform/sun4u/cfgadm/sbd.so.1

プラグインライブラリモジュール

/dev/cfg/sbd/slot*

記号名

/usr/sbin/cfgadm

cfgadm コマンド

属性

以下の属性については、attributes(5) を参照してください。

属性の種類	属性の値
Availability	SUNWkvm.u

関連項目

cfgadm(1M), devfsadm(1M), ifconfig(1M), mount(1M), pbind(1M), psradm(1M), psrinfo(1M), config_admin(3CFGADM), attributes(5)

注意事項

ここでは、メモリー削除操作の進行状況および cfgadm コマンドのプラットフォーム固有動作を監視する方法について説明します。

[メモリー削除の監視]

次のシェルスクリプトは、メモリー削除操作の進行状況を監視するために使用できます。

```
# cfgadm -c unconfigure -y sbd/slot0::memory &
#!/bin/sh
while true
do
    eval `cfgadm -l -s noheadings,cols=info -o parsable \
sbd/slot15.0::memory`
    if [ -n "$remaining" ]
    then
        echo $remaining mbytes
    else
        echo memory delete is done
        exit 0
    fi
    sleep 1
done
```

[Sun Fire 15000 プラットフォームにおける注意事項]

-t オプションと -x オプションは、**Sun Fire 15000** プラットフォーム上では違った動作をします。次に、その 2 つのオプションの動作を示します。

-t

システムコントローラが、CPU を使い、LPOST とその後に hpost コマンドを実行して、システムボードをテストします。I/O ボードのテストの場合は、ドライバが、-t オプションに答えてテストを始めると、テストは、自動的にユーザーの操作介入なしに実行されます。ドライバは、CPU の構成およびある範囲の隣接物理メモリーの構成を解除します。その後、システムコントローラにコマンドを送信し、ボードをテストします。システムコントローラが、トランザクション/エラーケージの内側から CPU とメモリーを使用して、I/O ボードをテストします。

-x assign | unassign

Sun Fire 15000 システム管理モデルでは、プラットフォーム管理者が、システムコントローラからプラットフォームハードウェアを制御します。プラットフォーム管理者だけが、ドメインの使用可能構成要素リストにボードを追加することによって、ドメインとの間で、未使用のボードを割り当てまたは割り当ての解除を行うことができます。ドメイン管理者は、ボードがすでにドメインの使用可能構成要素リスト中に存在する場合にかぎり、そのドメインに対してボードの割り当てまたは割り当て解除を行うことができます。

Sun Fire 15000 プラットフォームの場合、論理システムスロットは、*slot_nameX.Y* と表されます。ここで、*X* は拡張位置 (0 ~ 17) を表し、*Y* はスロット番号 (0 ~ 1) を表します。

次の例では、ドメインには、3 枚の CPU ボードまたはメモリーボード、および 2 枚の I/O ボードが含まれています。

```
# cfgadm -l -s "select=class(sbd)"
Ap_Id      Type      Receptacle  Occupant    Condition
sbd/slot11.0 CPU       connected   configured  ok
sbd/slot11.1 CPU       connected   configured  ok
sbd/slot15.0 CPU       connected   configured  ok
sbd/slot15.1 hpci     connected   configured  ok
```

名前	envmond — 環境モニターデーモン
形式	/usr/platform/SUNW,UltraSPARC-III-Netrtract/lib/envmond/ sparcv9/envmond [-d] [-f ファイル] [-g 頻度]
使用条件	SUNWcteux
機能説明	<p>envmond デーモンはシステム環境監視デバイスをポーリングし、調整行為が必要とされるような状態を調べます。そのためにデーモンは Solaris の初期起動時に構成ファイルを読み込み、どの環境デバイスを監視するかを調べます。各構成ファイルのエントリにはポリシーとして参照される環境デバイスが記述されており、サポートされているポリシーのエントリは <code>envmond.conf(4)</code> ファイルに記述されています。</p> <p>envmond デーモンは <code>syslogd(1M)</code> を経由してシステムログファイルにメッセージを記録します。</p> <p>envmond デーモンは <code>SIGHUP</code> シグナルを受け取ると構成情報ファイルを再読み込みします。</p>
オプション	<p><code>-d</code> デバッグモードに設定します。envmond はデーモンとしてではなく、フォアグラウンドで実行され標準入出力を引き継ぎます。エラーおよび警告メッセージは <code>syslogd(1M)</code> には記録されず、標準出力に書き出されます。</p> <p><code>-f ファイル</code> 代替の構成ファイルのパスを指定します。</p> <p><code>-g 頻度</code> ポーリングの間隔を指定します。デフォルトは 10 秒です。</p>
ファイル	<p>/usr/platform/SUNW,UltraSPARC-III-Netrtract/lib/envmond/ sparcv9/envmond</p> <p>実行可能デーモン</p> <p>/usr/platform/SUNW,UltraSPARC-III-Netrtract/lib/envmond/ sparcv9/*.so</p> <p>envmond ポリシー</p>

	<code>/platform/SUNW,UltraSPARC-III-Netractor/lib/envmond.conf</code>
	envmond 構成ファイル
関連項目	syslogd(1M), envmond.conf(4)
注意事項	envmond ポリシーはシステムの I2C デバイスから環境情報を取得します。 このデーモンはプロトタイプのため予告なく変更する可能性があります。

名前	nf_fddidaemon — PF FDDI SMT/SNM デーモン、およびこれに関連する処理の開始・停止
形式	nf_fddidaemon <i>start</i> / <i>stop</i>
使用条件	このコマンドは製品 SunFDDI がインストールされている場合のみ使用可能です。
機能説明	nf_fddidaemon スクリプトは、SNM デーモンおよびこれに関連する処理を開始または停止します。
オプション	<i>start</i> SNM デーモンを開始します。 <i>stop</i> SNM デーモンを停止します。 このコマンドを実行できるのはルートのみです。
関連項目	nf_snmd(1M)

名前	nf_install_agents — SunFDDI 用の SunNet Manager エージェントのインストール
形式	nf_install_agents
使用条件	このコマンドは製品 SunFDDI がインストールされている場合のみ使用可能です。
機能説明	<p>nf_install_agents スクリプトは、標準エージェントをインストールしているディレクトリに FDDI スキーマファイルをコピーし、SunNet Manager の構成ファイルを更新します。</p> <p>nf_install_agents コマンドに引数はありません。</p> <p>このコマンドを実行できるのはルートのみです。</p>
関連項目	nf_snmd(1M)

名前	nf_macid — 特定の nf (SunFDDI) インタフェースからの MAC アドレスの取得
形式	nf_macid <i>interface</i>
使用条件	このコマンドは、 <i>SunFDDI</i> 製品でのみ使用することができます。
機能説明	<p>このコマンドは、nf インタフェースに関連付けられている SunFDDI SBus カード上の IDPROM に照会し、格納されている MAC アドレスを取得します。MAC アドレスは、広域的に一意的 48 ビットアドレスで、Ethernet アドレスが取り出されるプールと同一のプールから取り出されます。</p> <p>nf_macid コマンドでは、MAC アドレスを SBus カード上またはインタフェースに設定することはできません。nf_macid で取得した MAC アドレスを SunFDDI インタフェースに割り当てるには、ifconfig に ether 引数を指定します。</p> <p>通常、マシン上のすべてのネットワークインタフェースには、ホストに存在する MAC アドレスを使用します。nf_macid で取得した MAC アドレスは、特別な状況でのみ使用します。</p> <p>nf_macid コマンドは、通常のユーザー (スーパーユーザー以外) が実行することができます。</p>
オプション	<i>interface</i> FDDI インタフェース (nf<num>) を指定します。このオプションは省略可能です。このオプションを省略すると、デフォルトで nf0 になります。
使用例	<p>次のコマンドは、nf0 の MAC アドレスを取得します。</p> <pre>% nf_macid 8:0:20:3e:da:5</pre> <p>次のコマンドは、nf0 インタフェースに SBus カードの IDPROM の MAC アドレスを設定します。</p> <pre># ifconfig nf0 ether 'nf_macid'</pre> <p>前のコマンドの後に ifconfig コマンドを使用し、nf0 に IP アドレスを割り当て、インタフェースを起動します。通常、このような ifconfig コマンドは起動ファイルから実行されます。</p>
関連項目	ifconfig (1M)

名前	nf_smtmon — SMT の監視
形式	nf_smtmon [-i <i>interface</i>] [-x] [-h <i>frametype</i>]
使用条件	このコマンドは、 SunFDDI 製品でのみ使用することができます。
機能説明	nf_smtmon は、受信した SMT フレームを表示するために使用します。コンソールが SMT MIB 情報の要求に対する応答を受信しない場合は、 FDDI プロキシシステムでこのコマンドを実行してください。
オプション	<p>このコマンドを実行するには、スーパーユーザーになる必要があります。</p> <p>-i interface FDDI インタフェース (SunFDDI の場合は nfnum) を指定します。このオプションを省略すると、すべての FDDI インタフェースについて、受信したフレームが表示されます。</p> <p>-x 受信したフレームを 16 進数で表示します。</p> <p>-h このコマンドの使用法を表示します。</p> <p>frametype 表示する SMT フレームの種類を 1 つまたは複数指定します。このオプションを省略すると、すべての種類のフレームが表示されます。指定することができるフレームの種類を以下に示します。</p> <p>ecf エコーフレーム。要求フレームと応答フレームを使用して、FDDI リングの SMT 間ループバックテストを行います。</p> <p>esf 拡張サービスフレーム。要求、応答、通知フレームを使用して、新しい SMT サービスが拡張されます。</p> <p>nif 隣接情報フレーム。要求、応答、通知フレームを使用して、ステーションアドレスと記述内容を通信します。</p> <p>pmf_get パラメタ管理フレーム (PMF) 取得要求。要求フレームと応答フレームを使用して、SMT 管理情報ベース (MIB) の属性値を取り出します。</p>

- rdf** 要求否定フレーム (応答のみ)。対応していない要求、あるいは不明な要求に対する応答として送信されます。
- sifconfig** 状態情報フレーム (**SIF**) 構成。要求フレームと応答フレームを使用して、リング上のステーションに対する構成情報を取り出します。
- sifoperation** 状態情報フレーム (**SIF**) 処理。要求フレームと応答フレームを使用して、リング上のステーションに対する処理情報を取り出します。
- srf** 状態報告フレーム。通知フレームを使用して、ステーションの状態を報告します。現在のバージョンの **SMT** デーモンは、**SRF** を送信しません。ただし、受信した **SRF** はすべてトラップとして **SNM** に渡されます。

使用例

```
nf_smtmon -i nf0 nif sifconfig
```

このコマンドは、nf0 (**SunFDDI**) インタフェースが受信した **NIF** および **SIF** 構成フレームを **16** 進数以外の形式で表示します。

```
nf_smtmon -i nf1 -x ecf
```

このコマンドは、nf1 (**SunFDDI**) インタフェースが受信した **ECF** フレームを **16** 進数形式で表示します。

関連項目

smt d(1M)

名前	nf_snmd — SunNet Manager デーモンに対するステーション管理 (SMT) の起動
形式	nf_snmd [-d] [-v5]
使用条件	このコマンドは、 <i>SunFDDI</i> 製品でのみ使用することができます。
機能説明	<p>起動時に、SNM デーモンはステーション管理プロセスを起動します。ステーション管理プロセスは、ステーションが SMT プロトコルを使用して他のステーションと通信したり、FDDI 統計情報を収集して、その情報を SunNet Manager (SNM) コンソールに返したりすることを可能にします。また、SNM デーモンは、SMT 要求と SMT 応答を受信したり、SNM に代わってリング上の他のステーションに SMT 要求を送信したりします。また、SMT デーモンは、受信した状態報告フレーム (SRF) をトラップの形式で SNM 管理ステーションに転送します。</p> <p>SNM デーモンによって起動されるプロセスには、ローカルエージェント (fdi) とプロキシエージェント (fddisnt) の 2 つの SNM エージェントがあります。他の SNM エージェントと同様に、ローカルエージェントとプロキシエージェントは、RPC を使用して SNM 管理ステーションと通信します。SNM 要求に対する応答時に、ローカルエージェントは、ローカルマシンで収集された FDDI 統計情報を返します。この統計情報は、nf_stat および nf_stat -m コマンドで表示される情報と同じです。</p> <p>プロキシエージェントは、SNM コンソールに、実際の SMT フレーム (ECF、ESF、NIF、SIF 構成、SIF 処理) と選択されている SMT MIB グループに対する属性値の 2 種類の SMT 情報を返すことができます。このエージェントは、SMT 要求フレームを発行し、SMT 応答フレームを受信することによってターゲットステーションから情報を収集します。</p> <p>ターゲットステーションから MIB 属性値を取り出すときには、PMF 取得要求フレームと応答フレームを使用します。ターゲットステーションが PMF 取得フレームに対応していない場合は、ターゲットステーションはプロキシシステムに RDF 応答を返します。コンソールからの MIB 属性値の要求が成功しない場合は、プロキシシステム上で SMT モニターを実行して、ターゲットステーションから RDF フレームが受信されたかどうかを調べてください。ターゲットステーションが PMF 取得フレームに対応していない場合は、NIF、SIF 構成、SIF 処理のいずれかのフレームを使用して、必要とする属性値を返すことができます。</p>

SMT MIB 属性グループの MAC、PATH、PORT には、インデックスパラメタが含まれます。これらのグループのいずれかの属性値に対するクイックダンプ要求をコンソールから発行すると、最初のインデックスに関連付けられている値のみが返されます (コンソールから見ると、この要求に関連付けられているキー値は 1 になります)。他のインデックスに関連付けられている属性値を調べるには、キーフィールドに目的のインデックスにキーフィールドが設定されたデータ報告要求を送信する必要があります。

ステーション上の /etc/opt/snm/snm.conf ファイルに変更 (na.fddi.trap-rendez エントリへのホスト名の追加など) を加えて、その変更を有効にするには、nf_snmd_kill によって SNM デーモンをいったん終了して、起動しなおす必要があります。

このコマンドを実行するには、スーパーユーザーになる必要があります。

オプション

-d (デバッグモード) ステーションが送信または受信したフレームごとに、nf_snmd が起動されたウィンドウに 1 行のエントリを表示します。このオプションを省略すると、エントリは表示されずにシステムプロンプトに戻ります。nf_snmd コマンドが /etc/rc2.d/S98nf_fddidaemon 以外に存在している場合、このオプションの使用はお勧めできません。

関連項目

nf_snmd_kill (1M), nf_stat(1M)

名前	nf_snmd_kill — ステーション管理 (SMT) SunNet Manager デーモンとその関連プロセスの終了
形式	nf_snmd_kill
使用条件	このコマンドは、 SunFDDI 製品でのみ使用することができます。
機能説明	<p>nf_snmd_kill スクリプトは、SNM デーモンとその関連プロセスを終了します。また、このコマンドは、SNM デーモンによって起動されたローカルエージェント (fddi) とプロキシエージェント (fddismt) の 2 つの SNM エージェントも終了します。このコマンドは、SNM デーモンが動作していないときは、使用しないでください。</p> <p>nf_snmd_kill コマンドは、引数を取りません。</p> <p>このコマンドを実行するには、スーパーユーザーになる必要があります。</p>
関連項目	nf_snmd(1M)

名前	nf_stat — SunFDDI インタフェースの統計情報の表示
形式	nf_stat [-m <i>interface</i>] [<i>interval</i>] [<i>count</i>]
使用条件	このコマンドは、 SunFDDI 製品でのみ使用することができます。
機能説明	<p>nf_stat ユーティリティーは、SunFDDI インタフェースの統計情報を表示します。この統計情報には、SunFDDI による ANSI FDDI 接続管理規格 (CMT) の実装に関する情報と、パケットスループットやステーションの隣接に関する情報があります。</p> <p>このユーティリティーは、パケットスループットの統計情報、再構成イベント、インタフェース例外を定期的に報告することができます。また、隣接ステーションの識別情報とその PHY に関する情報、いくつかの FORMAC エラーカウンタ情報も報告します。ハートビート信号中、ハードウェアからのカウンタと状態変数のいくつかは、定期的にホストに渡されます。これらの統計情報は、-m オプションを付けずにこのコマンドを起動すると表示されます。interval 値を指定せずにコマンドを実行すると、累積統計情報が表示されません。interval 値を指定してコマンドを実行すると、各値について前回の表示からの差分が表示されます。</p>
オプション	<p>-m 現在の最隣接の情報と FDDI/S タイマーの設定をダンプします (後の説明を参照)。このオプションを使用した場合、interval および count 引数は無効になります。-m オプションを付けて nf_stat コマンドを実行するには、スーパーユーザーになる必要があります。</p> <p>interface 統計情報を取得する SunFDDI インタフェース (nfnum) を指定します。</p> <p>interval 統計情報を表示する間隔 (秒) を指定します。</p> <p>count 統計情報を表示する回数を指定します。このオプションを省略すると、このユーティリティーは、^C (Control-C) によって終了させられるまで統計情報を表示し続けます。</p>
使用法	隣接ステーションに関する情報を表示するには、-m オプションを指定して nf_stat を起動します。表形式で次のカテゴリのデータが表示されます。

PhyA	<p>SunFDDI Dual が稼働しているマシンで、PHYA に接続されている隣接ステーションの PHY のタイプを情報を示します。値は、A、B、S、M、None (接続なし) のいずれかです。</p> <p>SunFDDI SAS (Single Attached Station) が稼働しているマシンでは、この欄は表示されません。『ANSI/FDDI Station Management (SMT)』 Rev7.2 (1992 年 6 月 25 日) の第 9 章を参照してください。</p>
PhyB	<p>SunFDDI Dual が稼働しているマシンで、PHYB に接続されている隣接ステーションの PHY のタイプを情報を示します。値は、A、B、S、M、None (接続なし) のいずれかです。</p> <p>SunFDDI SAS が稼働しているマシンでは、この欄は表示されません。『ANSI/FDDI Station Management (SMT)』 Rev7.2 (1992 年 6 月 25 日) の第 9 章を参照してください。</p>
PhyS	<p>SunFDDI SAS が稼働しているマシンで、PHYS に接続されている隣接ステーションの PHY のタイプを情報を示します。値は、A、B、S、M、None (接続なし) のいずれかです。端末集配装置に接続されている場合、値は M になります。</p> <p>SunFDDI Dual が稼働しているマシンでは、この情報は表示されません。</p>
Frame	<p>FDDI MAC の標準カウンタの 1 つで、受信されたフレームを示します。</p>
Error	<p>FDDI MAC の標準カウンタの 1 つで、このステーションで最初に E ビットが検出されたフレームを示します。</p>
Lost	<p>受信が中止されたフレームを示します。</p>
SA	<p>MAC アドレス。SunFDDI インタフェースの一意の 48 ビットアドレスです。IP ホスト名が存在する場合は、そのホスト名が表示されます。存在しない場合は、48 ビットの MAC アドレスが使用されます。</p>
UNA	<p>SMT NIF プロトコルを使用している、このステーションの上流の隣接ステーションのアドレスです。</p>
DNA	<p>SMT NIF プロトコルを使用している、このステーションの下流の隣接ステーションのアドレスです。</p>

状態情報を表示するには、-m オプションを指定せずに `nf_stat` を起動するか、`interface` または `interval` 値を指定して `nf_stat` を起動します。`interval` 値を指定せずにコマンドを実行すると、累積統計情報が表示されます。`interval` 値を指定してコマンドを実行すると、各値について前回の表示からの差分が表示されます。

-m オプションなしで `nf_stat` を使用する目的の 1 つは、Ring_OP (**Ring Operational**) を監視することです。1 秒あたりの `ring_op` が複数ある場合は、媒体に問題があり、解決する必要があります。

-m オプションを指定せずに起動すると、表形式で次のカテゴリのデータが表示されます。

Ring	リングが動作中と停止中 (すなわち、 Claim が成功した) のどちらであるかを示します。
------	---

注 - 次の 5 つのフィールドの意味については、SMT のマニュアルの第 9 章を参照してください。

(`ec_state`)。ECM 状態マシンの現在の状態を示します。有効な値は、Out、In、Trace、Leave、Path_Test、Insert、Check、Deinsert のいずれかです。

RMT	(<code>rmt_state</code>)。RMT 状態マシンの現在の状態を示します。有効な値は、Isolated、Non_Op、Ring_Op、Detect、Non_Op_Dup、Ring_Op_Dup、Directed、Rm_Trace のいずれかです。
-----	--

PCMA/PCMB (SunFDDI Dual の場合) PCMS (SunFDDI SAS の場合)

(`pc_state`)。PCM から他の管理実態に渡される、PCM 状態マシンの現在の状態情報を含む変数です。現在の有効な値は、Off (O)、Break (B)、Reject (R)、Connect (C)、Next (N)、Signal (S)、Join (J)、Verify (V)、Active (A)、Maint (M) のいずれかです。

Ring_OP	(Ring Operational)。リングが起動した回数 (すなわち、リングが停止した回数) を示します。
---------	--

XmitP	送信されたパケット数を示します。
-------	------------------

nf_stat(1M)

保守コマンド

関連項目

RecvP 受信されたパケット数を示します。
netstat(1M)

名前	nf_sync — 同期モードで操作する際の SunFDDI インタフェースの構成	
形式	nf_sync <i>nf<inst></i> [<i>tsync sap</i>]	
使用条件	このコマンドは製品 SunFDDI がインストールされている場合のみ使用可能です。	
機能説明	nf_sync ユーティリティーは、同期モードで操作する際に SunFDDI インタフェースを構成するために使用します。デフォルトでは、 SunFDDI インタフェースは非同期モードのトラフィックのみを転送するように構成されています。	
オプション	<i>nf<inst></i>	FDDI インタフェースを指定します。
	<i>tsync</i>	同期タイマーをナノ秒単位で指定します（最小値は 400000 ナノ秒）。
	<i>sap</i>	同期モードでサービスアクセスポイント (SAP) を使用することを指定します。
使用方法	<i>tsync</i> と <i>sap</i> に値を指定しないで nf_sync を実行した場合は、インタフェースの現在の構成が返されます。 SAP を非同期モードに構成し直す場合は、 tsync=0 と指定します。	
使用例	nf_sync nf0	nf0 (SunFDDI) インタフェースの現在の構成を表示します。
	nf_sync nf0 1000000 800	クロックレート 1000000 ナノ秒 (1ms) で同期モード転送を行うよう SAP 800 を構成します。

名前	pf_fddidaemon — PF FDDI SMT/SNM デーモン、およびこれに関連する処理の開始・停止
形式	pf_fddidaemon <i>start</i> / <i>stop</i>
使用条件	このコマンドは製品 SunFDDI がインストールされている場合のみ使用可能です。
機能説明	pf_fddidaemon スクリプトは、SNM デーモンおよびこれに関連する処理を開始または停止します。
オプション	<i>start</i> SNM デーモンを開始します。 <i>stop</i> SNM デーモンを停止します。 このコマンドを実行できるのはルートのみです。
関連項目	pf_snmd(1M)

保守コマンド

pf_install_agents(1M)

名前	pf_install_agents — SunFDDI 用の SunNet Manager エージェントのインストール
形式	pf_install_agents
使用条件	このコマンドは製品 <i>SunFDDI</i> がインストールされている場合のみ使用可能です。
機能説明	<p>pf_install_agents スクリプトは、標準エージェントをインストールしているディレクトリに FDDI スキーマファイルをコピーし、SunNet Manager の構成ファイルを更新します。</p> <p>pf_install_agents コマンドに引数はありません。</p> <p>このコマンドを実行できるのはルートのみです。</p>
関連項目	pf_snmd (1M)

名前	pf_macid — 特定の pf (SunFDDI/P) インタフェースからの MAC アドレスの取得
形式	pf_macid <i>interface</i>
使用条件	このコマンドは、 <i>SunFDDI/P</i> 製品でのみ使用することができます。
機能説明	<p>このコマンドは、pf インタフェースに関連付けられている SunFDDI PCI カード上の IDPROM に照会し、格納されている MAC アドレスを取得します。MAC アドレスは、広域的に一意の 48 ビットアドレスで、Ethernet アドレスが取り出されるプールと同一のプールから取り出されます。</p> <p>pf_macid コマンドでは、MAC アドレスを PCI カード上またはインタフェースに設定することはできません。pf_macid で取得した MAC アドレスを SunFDDI インタフェースに割り当てるには、ifconfig に ether 引数を指定します。</p> <p>通常、マシン上のすべてのネットワークインタフェースには、ホストに存在する MAC アドレスを使用します。pf_macid で取得した MAC アドレスは、特別な状況でのみ使用します。</p> <p>pf_macid コマンドは、通常のユーザー (スーパーユーザー以外) が実行することができます。</p>
オプション	<i>interface</i> FDDI インタフェース (pf<num>) を指定します。このオプションは省略可能です。このオプションを省略すると、デフォルトで pf0 になります。
使用例	<p>次のコマンドは、pf0 の MAC アドレスを取得します。</p> <pre>% pf_macid 8:0:20:3e:da:5</pre> <p>次のコマンドは、pf0 インタフェースに PCI カードの IDPROM の MAC アドレスを設定します。</p> <pre># ifconfig pf0 ether 'pf_macid'</pre> <p>前のコマンドの後に ifconfig コマンドを使用し、pf0 に IP アドレスを割り当て、インタフェースを起動します。通常、このような ifconfig コマンドは起動ファイルから実行されます。</p>
関連項目	ifconfig(1M)

名前	pf_smtmon — SMT の監視
形式	pf_smtmon [-i <i>interface</i>] [-x] [-h] [<i>frametype</i>]
使用条件	このコマンドは、 SunFDDI/P 製品でのみ使用することができます。
機能説明	pf_smtmon は、受信した SMT フレームを表示するために使用します。コンソールが SMT MIB 情報の要求に対する応答を受信しない場合は、 FDDI プロキシシステムでこのコマンドを実行してください。 このコマンドを実行するには、スーパーユーザーになる必要があります。
オプション	<p>-i <i>interface</i> FDDI インタフェース (SunFDDI/P の場合は pfnum) を指定します。このオプションを省略すると、すべての FDDI インタフェースについて、受信したフレームが表示されます。</p> <p>-x 受信したフレームを 16 進数で表示します。</p> <p>-h このコマンドの使用法を表示します。</p> <p><i>frametype</i> 表示する SMT フレームの種類を 1 つまたは複数指定します。このオプションを省略すると、すべての種類のフレームが表示されます。指定することができるフレームの種類を以下に示します。</p> <p>ecf エコーフレーム。要求フレームと応答フレームを使用して、FDDI リングの SMT 間ループバックテストを行います。</p> <p>esf 拡張サービスフレーム。要求、応答、通知フレームを使用して、新しい SMT サービスが拡張されます。</p> <p>nif 隣接情報フレーム。要求、応答、通知フレームを使用して、ステーションアドレスと記述内容を通信します。</p> <p>pmf_get パラメタ管理フレーム (PMF) 取得要求。要求フレームと応答フレームを使用して、SMT 管理情報ベース (MIB) の属性値を取り出します。</p>

- rdf** 要求否定フレーム (応答のみ)。対応していない要求、あるいは不明な要求に対する応答として送信されます。
- sifconfig** 状態情報フレーム (**SIF**) 構成。要求フレームと応答フレームを使用して、リング上のステーションに対する構成情報を取り出します。
- sifoperation** 状態情報フレーム (**SIF**) 処理。要求フレームと応答フレームを使用して、リング上のステーションに対する処理情報を取り出します。
- srf** 状態報告フレーム。通知フレームを使用して、ステーションの状態を報告します。現在のバージョンの **SMT** デーモンは、**SRF** を送信しません。ただし、受信した **SRF** はすべてトラップとして **SNM** に渡されます。

使用例

```
pf_smtmon -i pf0 nif sifconfig
```

このコマンドは、pf0 (SunFDDI/P) インタフェースが受信した **NIF** および **SIF** 構成フレームを 16 進数以外の形式で表示します。

```
pf_smtmon -i pf1 -x ecf
```

このコマンドは、pf1 (SunFDDI/P) インタフェースが受信した **ECF** フレームを 16 進数形式で表示します。

関連項目

smt(1M)

名前	pf_snmd — SunNet Manager デーモンに対するステーション管理(SMT) の起動
形式	pf_snmd [-d] [-v5]
使用条件	このコマンドは、 <i>SunFDDI/P</i> 製品でのみ使用することができます。
機能説明	<p>起動時に、SNM デーモンはステーション管理プロセスを起動します。ステーション管理プロセスは、ステーションが SMT プロトコルを使用して他のステーションと通信したり、FDDI 統計情報を収集して、その情報を SunNet Manager (SNM) コンソールに返したりすることを可能にします。また、SNM デーモンは、SMT 要求と SMT 応答を受信したり、SNM に代わってリング上の他のステーションに SMT 要求を送信したりします。また、SMT デーモンは、受信した状態報告フレーム (SRF) をトラップの形式で SNM 管理ステーションに転送します。</p> <p>SNM デーモンによって起動されるプロセスには、ローカルエージェント (fddi) とプロキシエージェント (fddisnt) の 2 つの SNM エージェントがあります。他の SNM エージェントと同様に、ローカルエージェントとプロキシエージェントは、RPC を使用して SNM 管理ステーションと通信します。SNM 要求に対する応答時に、ローカルエージェントは、ローカルマシンで収集された FDDI 統計情報を返します。この統計情報は、pf_stat および pf_stat -m コマンドで表示される情報と同じです。</p> <p>プロキシエージェントは、SNM コンソールに、実際の SMT フレーム (ECF、ESF、NIF、SIF 構成、SIF 処理) と選択されている SMT MIB グループに対する属性値の 2 種類の SMT 情報を返すことができます。このエージェントは、SMT 要求フレームを発行し、SMT 応答フレームを受信することによってターゲットステーションから情報を収集します。ターゲットステーションから MIB 属性値を取り出すときには、PMF 取得要求フレームと応答フレームを使用します。</p> <p>ターゲットステーションが PMF 取得フレームに対応していない場合は、ターゲットステーションはプロキシシステムに RDF 応答を返します。コンソールからの MIB 属性値の要求が成功しない場合は、プロキシシステム上で SMT モニターを実行して、ターゲットステーションから RDF フレームが受信されたかどうかを調べてください。ターゲットステーションが PMF 取得フレームに対応してしていない場合は、NIF、SIF 構成、SIF 処理のいずれかのフレームを使用して、必要とする属性値を返すことができます。</p>

SMT MIB 属性グループの MAC、PATH、PORT には、インデックスパラメタが含まれます。これらのグループのいずれかの属性値に対するクイックダンプ要求をコンソールから発行すると、最初のインデックスに関連付けられている値のみが返されます (コンソールから見ると、この要求に関連付けられているキー値は 1 になります)。他のインデックスに関連付けられている属性値を調べるには、キーフィールドに目的のインデックスにキーフィールドが設定されたデータ報告要求を送信する必要があります。

ステーション上の /etc/opt/snm/snm.conf ファイルに変更 (na.fddi.trap-rendez エントリへのホスト名の追加など) を加えて、その変更を有効にするには、pf_snmd_kill によって SNM デーモンをいったん終了して、起動しなおす必要があります。

このコマンドを実行するには、スーパーユーザーになる必要があります。

オプション

-d (デバッグモード) ステーションが送信または受信したフレームごとに、pf_snmd が起動されたウィンドウに 1 行のエントリを表示します。このオプションを省略すると、エントリは表示されずにシステムプロンプトに戻ります。pf_snmd コマンドが /etc/rc2.d/S98pf_fddidaemon 以外に存在している場合、このオプションの使用はお勧めできません。

関連項目

pf_snmd_kill(1M), pf_stat(1M)

名前	pf_snmd_kill — ステーション管理 (SMT) SunNet Manager デーモンとその関連プロセスの終了
形式	pf_snmd_kill
使用条件	このコマンドは、 <i>SunFDDI/P</i> 製品でのみ使用することができます。
機能説明	<p>pf_snmd_kill スクリプトは、SNM デーモンとその関連プロセスを終了します。また、このコマンドは、SNM デーモンによって起動されたローカルエージェント (fddi) とプロキシエージェント (fddismt) の 2 つの SNM エージェントも終了します。このコマンドは、SNM デーモンが動作していないときは、使用しないでください。</p> <p>pf_snmd_kill コマンドは、引数を取りません。</p> <p>このコマンドを実行するには、スーパーユーザーになる必要があります。</p>
関連項目	pf_snmd(1M)

名前	pf_stat — SunFDDI/P インタフェースの統計情報の表示
形式	pf_stat [-m] [<i>interface</i>] [<i>interval</i>] [<i>count</i>]
使用条件	このコマンドは、 SunFDDI/P 製品でのみ使用することができます。
機能説明	<p>pf_stat ユーティリティーは、SunFDDI/P インタフェースの統計情報を表示します。この統計情報には、SunFDDI/P による ANSI FDDI 接続管理規格 (CMT) の実装に関する情報と、パケットスループットやステーションの隣接に関する情報があります。</p> <p>このユーティリティーは、パケットスループットの統計情報、再構成イベント、インタフェース例外を定期的に報告することができます。また、隣接ステーションの識別情報とその PHY に関する情報、いくつかの FORMAC エラーカウンタ情報も報告します。ハートビート信号中、ハードウェアからのカウンタと状態変数のいくつかは、定期的にホストに渡されます。これらの統計情報は、-m オプションを付けずにこのコマンドを起動すると表示されます。interval 値を指定せずにコマンドを実行すると、累積統計情報が表示されます。interval 値を指定してコマンドを実行すると、各値について前回の表示からの差分が表示されます。</p>
オプション	<p>-m 現在の最隣接の情報と FDDI/S タイマーの設定をダンプします (後の説明を参照)。このオプションを使用した場合、interval および count 引数は無効になります。-m オプションを付けて pf_stat コマンドを実行するには、スーパーユーザーになる必要があります。</p> <p>interface 統計情報を取得する SunFDDI/P インタフェース(pfnum) を指定します。</p> <p>interval 統計情報を表示する間隔 (秒) を指定します。</p> <p>count 統計情報を表示する回数を指定します。このオプションを省略すると、このユーティリティーは、^C (Control-C) によって終了させられるまで統計情報を表示し続けます。</p>
使用法	隣接ステーションに関する情報を表示するには、 -m オプションを指定して pf_stat を起動します。表形式で次のカテゴリのデータが表示されます。

PhyA	<p>SunFDDI/P Dual が稼働しているマシンで、PHYA に接続されている隣接ステーションの PHY のタイプを情報を示します。値は、A、B、S、M、None (接続なし) のいずれかです。SunFDDI/P SAS (Single Attached Station) が稼働しているマシンでは、この欄は表示されません。『ANSI/FDDI Station Management (SMT)』 Rev7.2 (1992年 6 月 25 日) の第 9 章を参照してください。</p>
PhyB	<p>SunFDDI/P Dual が稼働しているマシンで、PHYB に接続されている隣接ステーションの PHY のタイプを情報を示します。値は、A、B、S、M、None (接続なし) のいずれかです。SunFDDI/P SAS が稼働しているマシンでは、この欄は表示されません。『ANSI/FDDI Station Management (SMT)』 Rev7.2 (1992 年 6 月 25 日) の第 9 章を参照してください。</p>
PhyS	<p>SunFDDI/P SAS が稼働しているマシンで、PHYS に接続されている隣接ステーションの PHY のタイプを情報を示します。値は、A、B、S、M、None (接続なし) のいずれかです。端末集配信装置に接続されている場合、値は M になります。SunFDDI/P Dual が稼働しているマシンでは、この情報は表示されません。</p>
Frame	<p>FDDI MAC の標準カウンタの 1 つで、受信されたフレームを示します。</p>
Error	<p>FDDI MAC の標準カウンタの 1 つで、このステーションで最初に E ビットが検出されたフレームを示します。</p>
Lost	<p>受信が中止されたフレームを示します。</p>
SA	<p>MAC アドレス。SunFDDI/P インタフェースの一意的 48 ビットアドレスです。IP ホスト名が存在する場合は、そのホスト名が表示されます。存在しない場合は、48 ビットの MAC アドレスが使用されます。</p>
UNA	<p>SMT NIF プロトコルを使用している、このステーションの上流の隣接ステーションのアドレスです。</p>
DNA	<p>SMT NIF プロトコルを使用している、このステーションの下流の隣接ステーションのアドレスです。</p>

状態情報を表示するには、`-m` オプションを指定せずに `pf_stat` を起動するか、`interface` または `interval` 値を指定して `pf_stat` を起動します。`interval` 値を指定せずにコマンドを実行すると、累積統計情報が表示されます。`interval` 値を指定してコマンドを実行すると、各値について前回の表示からの差分が表示されます。

`-m` オプションなしで `pf_stat` を使用する目的の 1 つは、Ring_OP (**Ring Operational**) を監視することです。1 秒あたりの `ring_op` が複数ある場合は、媒体に問題があり、解決する必要があります。

`-m` オプションを指定せずに起動すると、表形式で次のカテゴリのデータが表示されます。

Ring リングが動作中と停止中 (すなわち、**Claim** が成功した) のどちらであるかを示します。

注 - 次の 5 つのフィールドの意味については、**SMT** のマニュアルの第 9 章を参照してください。

ECM (`ec_state`)。ECM 状態マシンの現在の状態を示します。有効な値は、`Out`、`In`、`Trace`、`Leave`、`Path_Test`、`Insert`、`Check`、`Deinsert` のいずれかです。

RMT (`rmt_state`)。RMT 状態マシンの現在の状態を示します。有効な値は、`Isolated`、`Non_Op`、`Ring_Op`、`Detect`、`Non_Op_Dup`、`Ring_Op_Dup`、`Directed`、`Rm_Trace` のいずれかです。

PCMA/PCMB (SunFDDI/P Dual の場合) PCMS (SunFDDI/P SAS の場合)

(`pc_state`)。PCM から他の管理実態に渡される、PCM 状態マシンの現在の状態情報を含む変数です。現在の有効な値は、`Off (O)`、`Break (B)`、`Reject (R)`、`Connect (C)`、`Next (N)`、`Signal (S)`、`Join (J)`、`Verify (V)`、`Active (A)`、`Maint (M)` のいずれかです。

Ring_OP (**Ring Operational**)。リングが起動した回数 (すなわち、リングが停止した回数) を示します。

XmitP 送信されたパケット数を示します。

保守コマンド

pf_stat(1M)

関連項目

RecvP 受信されたパケット数を示します。
netstat(1M)

名前	rscadm — SUN™ Remote System Control (RSC) の管理	
形式	<pre> rscadm help rscadm resetrsc [-s] rscadm set <i>variable value</i> rscadm download [boot] <i>file</i> rscadm show [variable] rscadm date [-s] [[mmdd]HHMM mmddHHMM[cc]yy][.SS] rscadm send_event [-c] <i>message</i> rscadm modem_setup rscadm useradd <i>username</i> rscadm userdel <i>username</i> rscadm usershow [username] rscadm userpassword <i>username</i> rscadm userperm <i>username</i> [cuar] </pre>	
機能説明	<p>rscadm は SUN™ Remote System Control (RSC) を管理します。ホストサーバーが RSC と対話できるようにします。以下の操作がサポートされています:</p> <pre> rscadm help </pre>	<p>使用法の画面を表示します。</p> <pre> rscadm resetrsc </pre> <p>RSC をリセットします。実行できるリセットには、「ハード (hard)」リセットと「ソフト (soft)」リセットの 2 種類があります。デフォルトではハードリセットが実行されます。ソフトリセットは -s オプションを使用して選択できます。</p> <pre> rscadm set </pre> <p>RSC 設定変数を設定します。RSC 設定変数の例としては、RSC IP アドレスや RSC ホスト名があります。RSC 設定変数の全一覧は、RSC のマニュアルを参照してください。</p>

<code>rscadm download</code>	<p>RSC のファームウェアをプログラムします。ファームウェアには、起動監視と主イメージの 2 つの部分があります。デフォルトでは、rscadm は主ファームウェアイメージをダウンロードします。boot オプションは起動監視のプログラミングを選択します。</p>
<code>rscadm show</code>	<p>現在の RSC 設定変数の設定を表示します。変数が指定されないと、rscadm はすべての変数設定を表示します。</p>
<code>rscadm date</code>	<p>RSC の時刻と日付を、表示または設定します。</p> <p>-s オプションは RSC の時刻と日付をホストの時刻と日付に設定するために使用できます。</p>
<code>rscadm send_event</code>	<p>RSC にテキスト形式のイベントを送信します。RSC がイベント設定を基に、イベントを転送することがあります。</p>
<code>rscadm modem_setup</code>	<p>RSC モデムへの直接接続です。これはモデム設定のためにユーザーが AT コマンドを入力できるようにします。“~.” でプロンプトに戻ります。</p>
<code>rscadm useradd</code>	<p>ユーザーアカウントを RSC に追加します。RSC は独立した 4 ユーザーまでをサポートします。</p>
<code>rscadm userdel</code>	<p>ユーザーアカウントを RSC から削除します。</p>
<code>rscadm usershow</code>	<p>指定されたユーザーアカウントの詳細を表示します。ユーザー名が指定されないと、すべてのユーザーアカウントが表示されます。</p>
<code>rscadm userpassword</code>	<p>指定され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どのパスワードも上書きします。新</p>

しいパスワード設定の前に古いパスワードを
いっさい確認しません。**RSC** のマニュアルで
正しいパスワード形式を参照してください。

`rscadm userperm` ユーザー用の認証プロファイルを設定しま
す。詳細はマニュアルページの `userperm` の
説を参照してください。

オプション

以下のオプションが `rscadm` でサポートされています:

`rscadm resetrsc`

`[-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 **RSC** ハードウェアを物理的にリセッ
トします。ソフトリセットでは、リセットをまねて、**RSC** ソ
フトウェアが起動ファームウェアに制御を移します。

`rscadm download`

`[boot]`

起動モニターのフラッシュの場所をプログラムします。フ
ラッシュの主な位置は、通常はプログラムされます。

`rscadm show`

`[variable]` 特定の変数の値を表示します。

`rscadm date`

`[-s]` ホストの時刻と日付を設定します。

`[[mmdd]HHMM | mmddHHMM[cc]yy][.SS]`

日付です。

`mm` - 月

`dd` - 日

`HH` - 時

`MM` - 分

`cc` - 西暦 (4 桁表示) の上 2 桁

`yy` - 西暦の下 2 桁 (十の位と一の位)

`SS` - 秒

rscadm send_event

[-c] 重要なイベントを送信します。-c なしだと、**send_event** は警告を送信します。警告は **RSC** イベントログにのみ記録され、他に転送されません。

rscadm usershow

[username] 情報を表示する **RSC** アカウント名です。ユーザー名の指定がなければ、すべてのアカウントが表示されます。

rscadm userperm

[cuar] **RSC** アカウントのアクセス権を設定します。アクセス権の指定がないと、4つのアクセス権すべてが使用不可になります。オプションには、ユーザーにコンソールの接続を許可 (**c**)、ユーザーに **RSC** アカウント修正のためのユーザーコマンド使用を許可 (**u**)、ユーザーに **RSC** 設定値の管理・変更を許可 (**a**)、ユーザーに **RSC** のリセットとホストの電源投入・切断を許可 (**r**) があります。

オペランド

以下のオペランドが **rscadm** でサポートされています:

rscadm set

variable 設定する **RSC** 設定変数です。設定変数の一覧は、**RSC** のマニュアルを参照してください。

value **RSC** 設定変数に設定する値です。正しい値の一覧は、**RSC** のマニュアルを参照してください。

rscadm download

file ダウンロードするファームウェアファイルです。ファイルの内容は **RSC** 起動監視のイメージか、**RSC** の主イメージである必要があります。

rscadm send_event

message イベントを記述するテキストメッセージです。引用符で囲んでください。

	<code>rscadm useradd</code>
	<i>username</i> 新しい RSC アカウント用のユーザー名です。
	<code>rscadm userdel</code>
	<i>username</i> 削除される RSC です。
	<code>rscadm userpassword</code>
	<i>username</i> パスワードを設定する RSC アカウントです。
	<code>rscadm userperm</code>
	<i>username</i> アクセス権を変更する RSC アカウントです。
終了状態	<code>= 0</code> 成功
	<code>!= 0</code> 失敗 (状態メッセージを表示)
使用例	<pre># rscadm date # rscadm date -s # rscadm date 050113101998 # rscadm set hostname rsc15 # rscadm show # rscadm show hostname # rscadm send_event -c "The UPS signaled a loss in power!" # rscadm send_event "The disk is close to full capacity" # rscadm useradd rscroot # rscadm userdel olduser # rscadm usershow # rscadm usershow rscroot # rscadm userperm rscroot cuar # rscadm userperm newuser c # rscadm userperm newuser</pre>
注意事項	<p><code>rscadm modem_setup - "~."</code> は改行後にのみ動作します。</p> <p><code>rscadm</code> を <code>root</code> になって実行する必要があります。</p>
制限事項	既知の問題点はありません。

名前	sunvts — SunVTS カーネルとユーザーインタフェースの呼び出し
形式	sunvts [-llepqrstv] [-o <i>option_file</i>] [-f <i>log_dir</i>] [-h <i>hostname</i>]
使用条件	SUNWvts
機能説明	sunvts コマンドは、 SunVTS のユーザーインタフェースとカーネルを同一のシステム上で呼び出すために使用されます。ローカルシステムでユーザーインタフェースを起動し、遠隔システムの SunVTS カーネルに接続することができます。デフォルトでは、 CDE 環境では CDE Motif グラフィックインタフェースを表示し、 OpenWindows 環境では OpenLook グラフィックインタフェースを表示し、ウィンドウ環境以外のシステムでは TTY インタフェースを表示します。デフォルトでは、 sunvts は、稼働中のオペレーティングシステム (32 ビットまたは 64 ビット) に対応する vtsk(1M) を起動します。
オプション	<p>-l SunVTS OpenLook グラフィックインタフェースを起動します。</p> <p>-e セキュリティの確認機能を無効にします。</p> <p>-f <i>log_dir</i> 代替のログファイルディレクトリ <i>log_dir</i> を指定します。デフォルトのログファイルディレクトリ <i>log_dir</i> は <code>/var/opt/SUNWvts/logs</code> です。</p> <p>-h <i>hostname</i> ローカルシステムで SunVTS ユーザーインタフェースを起動し、セキュリティの確認に成功した後、指定されたホストの SunVTS カーネルに接続、または呼び出します。</p> <p>-o <i>option_file</i> 指定された <i>option_file</i> から読み込んだテスト用オプションで SunVTS カーネルを起動します。デフォルトのオプションは <code>/var/opt/SUNWvts/options</code> に格納されています。</p> <p>-p テストシステムのデバイスを調査しないように指定して SunVTS カーネル vtsk(1M) を起動します。</p> <p>-q テストが停止した場合に、SunVTS カーネルとユーザーインタフェースの両方を自動的に終了します。</p> <p>-s 選択されたテストグループからテストを自動的に開始します。このフラグは、-o <i>option_file</i> フラグとともに使用する必要があります。</p>

	-t	CDE または OpenLook インタフェースの代わりに、TTY ベースのインタフェースである vtstty(1M) を起動します。
	-v	vtsui(1M) と vtsk(1M) のバージョン情報を表示します。
注意事項		vtsk(1M) がテストシステムですでに動作している場合、sunvts コマンドは、-e、-o、-f、-q、-p、-s オプションを無視します。
関連項目		vtsk(1M), vtstty(1M), vtsui(1M), vtsui.ol(1M), vtsprobe(1M)

名前	vts_cmd - SunVTS カーネル (vtsk) へのコマンドの送信
形式	vts_cmd [コマンド] [引数]
使用条件	SUNWvts
機能説明	<p>vts_cmd(1M) は、1 つのコマンドを SunVTS カーネル (vtsk) へ送信する UNIX シェルアプリケーションです。テスト対象のシステムの SunVTS カーネルは、標準出力に応答を送ります。</p> <p>SunVTS アプリケーションプログラミングインタフェース (API) は、文字ベースで設計されています。つまり、(コマンドの) 文字列を SunVTS カーネルに送信できるということです。このとき SunVTS カーネルも文字列の形式で応答します。</p> <p>vts_cmd では UNIX のコマンド行で、コマンドの発信や応答の受信を行うことができます。</p>
オプション	<p>vts_cmd は、以下にリストしたコマンドを使用します。いずれの場合でも vts_cmd に続けてコマンド (コマンドによっては引き数が必要) を指定します。使用方法については、「使用例」を参照してください。</p> <p>以下にリストしたコマンド記述の一部は、テストノードについて言及していません。SunVTS API には、システムを頂点として、各テストグループがその下に位置し、底辺には個々のテストがあるという、テストノードの階層があります。以下のコマンドの説明では、システムについては、スラッシュ("/") で表記します。テストグループは以下の名称のいずれかになります。それぞれ、Processor(s)、Memory、Network、SCSI-Devices(esp0)、Comm.Ports、Graphics、OtherDevices、その他のユーザーの指定したグループです。個々のテストは具体的なデバイス名やテスト名 (例: sound0(audio)) で表記します。</p> <p>list testnode</p> <p>指定した <i>testnode</i> 以下にあるすべてのテストノードの表示。</p> <p>config testnode</p> <p><i>testnode</i> で指定したテストノードの構成情報の表示。</p>

status [*testnode*] [-r]

システムのテスト状態情報の表示。**testnode** が指定されている場合は、そのテストノードの状態情報が表示されます。**-r** オプションを指定すると、すべてのテストノードの状態情報が再帰的に表示されます。

option [*testnode*] [-l] [-h|n|s|t|a]

いずれも、指定したテストノードに関連しているすべてのオプションを表示します。または、指定したオプションをテストノード中に設定します。

テストノードのオプションを表示するには、テストノードに続いてオプションを指定し、以下のカテゴリの 1 つを入力してください。

- h **Threshold**
- n **Notify category**
- s **Scheduling category**
- t **Test execution category**
- a **Advanced category**

vts_cmd は全オプション、さらには各オプションの設定を出力します。**-l** オプションを指定すると、オプションの内容を省略全設定の説明付きで表示されます。

option [*testnode*] [*test_option*] [-g|s|x|y|z]

- g このオプションは、指定されたテストのインスタンスへの現在のオプション設定を、同じグループにあるすべての別のグループにあるものには適用されません。
- s このオプションは、指定されたテストのインスタンスへの現在のオプション設定を、(-g オプションのようにグループに対してではなく) そのシステムにある同じテストすべてに対応する全インスタンスに適用します。

- x このオプションは、指定されたテストのインスタンスへの現在のオプション設定を、そのテストに属するすべてのインスタンスに適用します。
- y このオプションは、指定されたテストのインスタンスへの現在のオプション設定を、ある特定のグループに属するすべての同じテストの全インスタンスに適用します。
- z このオプションは、指定されたテストのインスタンスへの現在のオプション設定を、全システムのすべての同じテストの全インスタンスに適用します。

オプションを設定するには、*testnode* のすぐあとに *option* と *setting* を指定する必要があります。以下のような書式で使用します。

```
vts_cmd option testnode[option:setting]
```

オプションの変更が成功すると、*vts_cmd* によって "DONE" と表示されます。

select *testnode*

テストノードを選択します。テストノードが選択されると、そのテストノードに関連付けられている全てのテストが実行可能になり、テスト開始時に実行されます。

たとえば、**Graphics** テストノードを選択した場合、すべてのグラフィックス関連テストが実行可能になります。"**fpu(fputest)**" テストを選択した場合は、このテストだけ実行可能になります。

deselect *testnode*

テストノードを選択解除します。テストノードが選択解除されると、そのテストノードに関連付けられている全てのテストが実行不可能になり、テスト時に実行されなくなります。

たとえば、**OtherDevices** テストノードを選択解除した場合、**OtherDevices** に該当するすべてのテストが実行不可能になります。"**cgsix0(cg6)**" テストを選択した場合は、このテストだけ実行可能になります。

start

実行可能な (選択された) すべての **SunVTS** テストを開始します。

stop

実行中のすべての **SunVTS** テストを停止します。

suspend

実行中のすべての **SunVTS** テストを中断 (または一時停止) します。テストを再開したい場合は、"resume" と入力してください。

resume

中断中のテストを再開します。

reset

すべての **SunVTS** テストをリセットし、エラー回数をゼロにします。

probe

テスト中のシステム上の全デバイスを調べ、**SunVTS** カーネルのデバイスリストを更新します。

もしリスト上のデバイスが、この検査で確認できない場合、そのデバイスはリストから削除されます。反対に、リストにないデバイスが検査中に発見された場合、そのデバイスはリストに追加されます。

load *option_file*

オプションファイルを読み込みます。ファイルが読み込まれるとシステムやオプションの設定は、そのオプションファイルの設定に変更されます。

オプションファイルは、/var/opt/SUNWvts/options ディレクトリに保存されます。

store *option_file*

システム情報とテストオプションがリストされたオプションファイルを作成します。作成されたファイルは、/var/opt/SUNWvts/options ディレクトリに保存されます。

quit

SunVTS カーネル (vtsk) を終了します。

invokeds

テスト手順スケジューラを起動します。

quitds

テスト手順スケジューラを終了します。

loadseq *sequence_file*

シーケンスファイルを読み込みます。ファイルが読み込まれると、テスト手順スケジューラユーザーインタフェースに、シーケンスファイルに記録されたタスクが反映されます。

storeseq *sequence_file*

すべてのタスクがリストされたシーケンスファイルを作成して、`/var/opt/SUNWvts/sequences` ディレクトリに保存します。

statusseq

現在実行中のシーケンスの状態情報を含む文字列を返します。この文字列はコンマ (",") で区切られた 4 つのフィールドから構成されます。それぞれ、SunVTS の現在の状態、現在のシーケンスのループ回数、シーケンスの全ループ回数、現在実行中のタスクの位置です。

startseq

テスト手順スケジューラのタスク実行を開始します。

stopseq

シーケンスファイル中の現在実行中のタスクを停止します。再開時には、停止した時点のタスクから再実行されます。

resumeseq

シーケンスファイルの実行を再開します。シーケンスがリセットされていない限り、再開時には、停止した時点のタスクから再実行されます。シーケンスがリセットされた場合は、シーケンスファイルの先頭から実行が開始されます。

resetseq

シーケンスファイルの実行開始ポイントを先頭に設定します。またテストパスやエラー回数もリセットされます。

suspendseq

シーケンスファイル中の現在実行中のタスクを中断します。

removeseq *sequence_file*

/var/opt/SUNWvts/sequences ディレクトリにあるシーケンスファイルのリストから、***sequence_file*** で指定したファイルを削除します。

listtask

現在読み込まれているシーケンスファイル中のタスクのリストを表示します。

addtask *task_name* [*i*]

task_name で指定したタスクを、シーケンスファイルの ***i*** 番目の位置に追加します。指定した位置に達していない場合、タスクはリストの末尾に追加されます。

deletetask [*i*]

選択したシーケンスの指定した位置にあるタスクを削除します。

loadtask *task_name*

タスクファイルを読み込みます。ファイルが読み込まれると、システムやオプションの設定は、そのタスクファイルの設定リストに従って変更されます。

setloopcount *count*

現在実行中のシーケンスのループ回数を *count* に設定します。

getvtsmode

SunVTS カーネルの現在のモード情報を取得します。

使用例

テスト中のシステムの構成情報を表示するには、以下のように config コマンドを使用します。

```
sample% vts_cmd config /
/[Hostname:sample,Model:SPARCstation-10,SunVTS version:1.0]:idle
```

オプションファイルを読み込むには、以下のようにして load コマンドを使用します。

```
sample% ls /var/adm/sunvtslog/options
CPU_options          sample          options
sbus_standard
```

```
sample% vts_cmd load sbus_standard
DONE
```

Comm.Ports テストノードのすべてのシステムオプションを印刷するには、option コマンドを使用して、以下のようにパイプでローカルプリンタに出力してください。

```
sample% vts_cmd option Comm.Ports -l | lp
request id is printer-213 (standard input)
```

環境変数

VTS_CMD_HOST=*hostname*

SunVTS カーネル (*vtsk*) が実行されているテスト中のシステムのホスト名です。この環境変数が設定されていない場合、vts_cmd はローカルシステムの SunVTS カーネルへのコマンドの送信を試みます。

関連項目

『SunVTS ユーザーマニュアル』

名前	vtsk — SunVTS 診断カーネル
形式	vtsk [-epqsv] [-o <i>options_file</i>] [-f <i>logfile_directory</i>]
使用条件	SUNWvts
機能説明	<p>vtsk コマンドは、バックグラウンドプロセスとして SunVTS 診断カーネルを起動します。一度に動作する vtsk は 1 つだけです。スーパーユーザーだけが、このコマンドを実行することができます。</p> <p>vtsk は、通常、sunvts(1M) によって自動的に起動されます (他に動作中の vtsk がない場合)。また、vtsk は、vtsui または vtsui.ol から接続要求がある場合に、inetd(1M) によって呼び出されます。この場合、vtsui(1M) または vtsui.ol(1M) によって指定されたターゲットホストの vtsk を実行する前に、セキュリティーファイル .sunvts_sec のアクセス権が確認されます。</p> <p>SunVTS 3.0 では、64 ビットの vtsk が SUNWvtsx パッケージに含まれています。この vtsk は、デフォルトで /opt/SUNWvts/bin/sparcv9 ディレクトリに他の 64 ビット SunVTS バイナリとともにインストールされます。SUNWvts パッケージに含まれている従来の 32 ビット vtsk は、デフォルトでは /opt/SUNWvts/bin ディレクトリにインストールされます。vtsk を通常呼び出す sunvts コマンドは、現在稼働している OS の形式 (32 ビットまたは 64 ビット) を判定し、対応する vtsk を起動します。</p> <p>/opt/SUNWvts/bin がデフォルトのインストール先ですが、パッケージのインストールプログラムで /opt/SUNWvts/bin の親ディレクトリとして別のディレクトリを指定することもできます。</p>
オプション	<p>-e すべての接続要求のセキュリティーを確認します。</p> <p>-p システム構成を調査しないで SunVTS 診断カーネルを開始します。</p> <p>-q テストが完了したときに、SunVTS 診断カーネルと接続されたユーザーインタフェースの両方を終了します。</p> <p>-s 有効になっているテストをただちに実行します。</p>

	-v	SunVTS 診断カーネルのバージョン情報を表示します。
	-o <i>options_file</i>	SunVTS 診断カーネルを開始し、 <i>options_file</i> で指定されたオプションファイルの内容によってテストオプションを設定します。
	-f <i>logfile_directory</i>	デフォルト以外の、代替ログファイルディレクトリを指定します。
終了状態	以下の終了値が返されます。	
	0	正常終了
	-1	異常終了
ファイル	/var/opt/SUNWvts/options	デフォルトのオプションファイルディレクトリ
	/var/opt/SUNWvts/logs	デフォルトのログファイルディレクトリ
関連項目	sunvts(1M), vtsui(1M), vtsui.ol(1M), vtstty(1M), vtsprobe(1M)	

名前	vtsprobe — SunVTS カーネルのデバイス調査情報の出力
形式	vtsprobe [-m] [-h <i>hostname</i>]
使用条件	SUNWvts
機能説明	vtsprobe は、SunVTS カーネルのデバイスおよび設定情報を表示するユーティリティです。出力には、SunVTS によってデバイスに割り当てられたグループ、デバイス名、デバイスインスタンス、デバイスに対応するテスト名、デバイス固有のテスト調査から得た設定情報が含まれます。
オプション	<p>-m 製造モードを特定します。製造モードは、スクリプトファイルを使用して読み取ることが容易な書式で調査情報を出力します。</p> <p>-h <i>hostname</i> 接続するホスト名を <i>hostname</i> に指定し、デバイス情報と設定情報を得ます。このオプションを指定しないと、現在のホストが指定されます。</p>
使用法	SunVTS カーネルを起動した後に、調査の出力を得るためにシェルプロンプトで vtsprobe を入力することができます。SunVTS の起動方法については、sunvts(1M) のマニュアルページを参照してください。
使用例	<p>sun4m SPARCclassic で vtsprobe を実行した場合の出力を以下に示します。</p> <pre>% vtsprobe Processor(s) system(system) System Configuration=sun4m SPARCclassic System clock frequency=50 MHz SBUS clock frequency=25 MHz fpu(fputest) Architecture=sparc Type=TI TMS390S10 or TMS390S15 microSPARC chip Memory kmem(vmem) Total: 143120KB mem(pmem) Physical Memory size=24 Mb SCSI-Devices(esp0)</pre>

```
c0t2d0(rawtest)
    Capacity: 638.35MB
    Controller: esp0
    Vendor: MICROP
    SUN Id: 1588-15MBSUN0669
    Firmware Rev: SNOC
    Serial Number: 1588-15MB103
c0t2d0(fstest)
    Controller: esp0
c0t3d0(rawtest)
    Capacity: 404.65MB
    Controller: esp0
    Vendor: SEAGATE
    SUN Id: ST1480    SUN0424
    Firmware Rev: 8628
    Serial Number: 00836508
c0t3d0(fstest)
    Capacity: 404.65MB
    Controller: esp0
    Vendor: SEAGATE
    SUN Id: ST1480    SUN0424
    Firmware Rev: 8628
    Serial Number: 00836508
c0t3d0(fstest)
    Controller: esp0
c0t6d0(cdtest)
    Controller: esp0
tapel(tapetest)
    Drive Type: Exabyte EXB-8500 8mm Helical
Scan
Network
    isdn0(isdntest)
        NT Port  TE Port
    le0(nettest)
        Host_Name: ctech84
        Host Address: 129.146.210.84
        Host ID: 8001784b
        Domain Name: scsict.Eng.Sun.COM
Comm.Ports
    zs0(sptest)
        Port a -- zs0 /dev/term/a : /devices/ ... a
```

```

Port b -- zs1 /dev/term/b : /devices/ ... b
Graphics
  cgthree0 (fbtest)
OtherDevices
  bpp0 (bpptest)
    Logical name: bpp0
  sound0 (audio)
    Audio Device Type: AMD79C30
  sound1 (audio)
    Audio Device Type: DBRI Speakerbox
  spd0 (spdtest)
    Logical name: spd0

```

注意事項

vtsprobe の出力は、SunVTS による調査が成功するように調査対象のデバイスがシステムに正しく構成されていることと、デバイス固有のテストがあることに依存します。

デバイスが正しく構成されていないか、デバイスに関連付けられた調査機能がない場合、vtsprobe はそのデバイスに関連する情報を出力することができません。

関連項目

sunvts(1M), vtsk(1M), vtsui(1M), vtsui.ol(1M), vtstty(1M)

名前	vtstty — SunVTS の TTY インタフェース
形式	vtstty [-qv] [-h <i>hostname</i>]
使用条件	SUNWvts
機能説明	vtstty は、非ウィンドウ環境における SunVTS のデフォルトのインタフェースです。システムのシリアルポートに接続された端末などの非ウィンドウ環境で使用することができます。ただし、vtstty の使用は、非ウィンドウ環境に限定されません。シェルウィンドウからも vtstty を使用することができます。
オプション	<p>-q SunVTS の終了条件が満たされたときに、自動的に終了します。</p> <p>-v vtstty のバージョンを表示します。このオプションを指定すると、インタフェースは起動されません。</p> <p>-h <i>hostname</i> <i>hostname</i> によって指定されたホスト上で動作している SunVTS カーネルに接続します。</p>
使用法	<p>vtstty 画面は、4 つのパネル (Main Control、Status、Test Groups、Console) によって構成されます。これらのパネルは、特定の機能を実行するための選択肢や情報を表示します。パネルがアスタリスクで囲まれ、現在の項目が強調表示されているときは、そのパネルは「操作対象になっている」または「選択されている」と言えます。パネルで項目を選択するには、まず、そのパネルを選択状態にする必要があります。</p> <p>パネルで選択することができる項目の種類を以下に示します。</p> <p>テキスト文字列 選択された場合、別のパネルを表示するか機能を実行する選択肢を記述します。たとえば「stop」は、SunVTS のテストを停止します。</p> <p>データ入力フィールド 数値またはテキストデータを入力または編集します。</p> <p>チェックボックス 「[]」のように表されます。チェックボックスは、対応している項目が選択されているかどうかを示します。チェックボックスは、選択されていない[] 選択されている [*] のいずれかになります。</p>

操作対象の移動、選択、他の機能の実行に関するキー割り当てを以下に示します。

TAB または <CTRL>W	操作対象を別のパネルへ移動します。
RETURN	現在の項目を選択します。
スペースバー	チェックボックスを切り替えます。
上矢印または <CTRL>U	1 項目上へ移動します。
下矢印または <CTRL>N	1 項目下へ移動します。
左矢印または <CTRL>P	1 項目左へ移動します。
右矢印または <CTRL>R	1 項目右へ移動します。
Backspace	データ入力フィールドのテキストを削除します。
ESC	表示を取り消します。
<CTRL>F	前方向へスクロールします (スクロール可能なパネルで)。
<CTRL>B	後方向へスクロールします (スクロール可能なパネルで)。
<CTRL>X	vtstty を終了しますが、SunVTS カーネルは動作を継続します。
<CTRL>L	画面を再描画します。

注意事項

1.telnetセッションから vtstty を実行するには、以下の手順に従います。

- a.telnetを起動する前に、「行」と「カラム」の値を決めます (stty(1) を参照)。
- b. telnetを実行した後に termを適切な形式に設定します (例: set term=vt100)
- c. カラムと行の値をあらかじめ決められた値に設定します (stty(1) を参照)。

保守コマンド

vtstty(1M)

関連項目

2. 端末の種類を記述する環境変数が正しく設定されていることを確認し、
vtstty を実行します。

sunvts(1M), vtsk(1M), vtsui(1M), vtsui.ol(1M), vtsprobe(1M)

名前	vtsui — SunVTS グラフィックユーザーインタフェース (CDE)
形式	vtsui [-qv] [-h <i>hostname</i>]
使用条件	SUNWvts
機能説明	<p>vtsui コマンドは、SunVTS グラフィックユーザーインタフェースの CDE Motif バージョンを起動します。実行中の vtsui には、同時に複数のインスタンスがあり、すべてが 1 つの SunVTS 診断カーネル vtsk(1M) へ接続されます。診断カーネル vtsk(1M) を実行しているホストマシン名は、グラフィックユーザーインタフェースウィンドウのタイトルバーに表示されます。</p> <p>vtsui は、sunvts(1M) コマンドによって自動的に起動されます。また、vtsui は、inetd(1M) が実行中である場合に、vtsk(1M) を起動するために使用することができます。その場合、ターゲットホスト上で vtsk を実行する前に、セキュリティーファイル (.sunvts_sec) のアクセス権が確認されます。グラフィックユーザーインタフェースの使用については、『SunVTS ユーザーマニュアル』を参照してください。</p>
オプション	<p>-q 評価が終了したら、SunVTS グラフィックユーザーインタフェースを終了します。</p> <p>-v グラフィックユーザーインタフェースのバージョン情報だけを表示します。</p> <p>-h <i>hostname</i> SunVTS グラフィックユーザーインタフェースを起動し、<i>hostname</i> 上で動作中の SunVTS 診断カーネルに接続します。SunVTS 診断カーネルが動作していない場合は、セキュリティーの確認後に SunVTS 診断カーネルを起動します。<i>hostname</i> を指定しないと、ローカルホストが指定されます。</p>
終了状態	<p>次の終了値が返されます。</p> <p>0 正常終了</p> <p>1 異常終了</p>
関連項目	sunvts(1M), vtsk(1M), vtsui.ol(1M), vtstty(1M), vtsprobe(1M)

名前	vtsui.ol — SunVTS グラフィックユーザーインタフェース (OpenLook)
形式	vtsui.ol [-qv] [-h <i>hostname</i>]
使用条件	SUNWvts
機能説明	<p>vtsui.ol コマンドは、SunVTS グラフィックユーザーインタフェースの OpenLook バージョンを起動します。実行中の vtsui.ol には、同時に複数のインスタンスがあり、すべてが 1 つの SunVTS 診断カーネル vtsk(1M) へ接続されます。診断カーネル vtsk(1M) を実行しているホストマシン名は、グラフィックユーザーインタフェースウィンドウのタイトルバーに表示されません。</p> <p>vtsui.ol は、inetd(1M) が実行中である場合に、vtsk(1M) を起動するために使用することができます。その場合、ターゲットホスト上で vtsk を実行する前に、セキュリティファイル (.sunvts_sec) のアクセス権が確認されます。vtsui.ol は、sunvts (1M) コマンドからも自動的に起動されます。グラフィックユーザーインタフェースの使用については、『SunVTS ユーザーマニュアル』を参照してください。</p>
オプション	<p>-q 評価が終了したら、SunVTS グラフィックユーザーインタフェースを終了します。</p> <p>-v グラフィックユーザーインタフェースのバージョン情報だけを表示します。</p> <p>-h <i>hostname</i> SunVTS グラフィックユーザーインタフェースを起動し、<i>hostname</i> 上で動作中の SunVTS 診断カーネルに接続します。SunVTS 診断カーネルが動作していない場合は、セキュリティの確認後に SunVTS 診断カーネルを起動します。<i>hostname</i> を指定しないと、ローカルホストが指定されます。</p>
終了状態	<p>次の終了値が返されます。</p> <p>0 正常終了</p> <p>1 異常終了</p>
サポートの中止について	<p>SunVTS OPEN LOOK ユーザーインタフェースでは、SunVTS の最新機能はサポートされません。また、Solaris オペレーティング環境で OPEN LOOK 環境が提供されなくなった際には、SunVTS OPEN LOOK ユーザーインタフェースは提供されなくなります。最新機能をすべて利用するには、SunVTS CDE イ</p>

関連項目

ンタフェースを使用してください。サポートの中止に関する最新の情報については、**Solaris** オペレーティング環境の『ご使用にあたって (SPARC 版)』マニュアルの「サポート中止に関する情報」を参照してください。

sunvts(1M), vtsk(1M), vtsui(1M), vtstty(1M), vtsprobe(1M)

第3章

ファイル形式

名前	envmond.conf — 環境モニターデーモン用構成ファイル
形式	/platform/SUNW,UltraSPARC-III-Netractor/lib/envmond.conf
機能説明	<p>envmond.conf ファイルは、システム環境モニターデーモン envmond 用の構成ファイルです。このデーモンはシステム環境上に存在する装置を監視し、対応が必要となる条件をチェックします。envmond デーモンは syslogd(1M) によって、システムログファイルにメッセージを記録します。</p> <p>構成ファイルの各エントリは共有オブジェクトライブラリに関する情報をデーモンに与えます。情報は装置を監視する際のポリシーとして参照されます。ポリシーの各エントリには同じ形式で、envmond デーモンとポリシー間のインタフェースを記述します。ポリシーは監視される装置だけを認識し、envmond.conf ファイルのポリシーのエントリには、policy-args フィールドに構成可能な変数を含めることができます。</p> <p>すべてのポリシーエントリは同じ形式になります。</p> <p style="text-align: center;"><i>poll-interval policy-name policy-args</i></p> <p>3つのフィールドはそれぞれ空白文字によって区切られます。<i>policy-args</i> を次の行に続けて記述する場合は、行末にバックスラッシュ (\) を使用します。</p> <p>envmond.conf ファイルの各フィールドの説明を以下に示します。</p> <p><i>poll-interval</i> ポリシーの check 関数を起動する間隔を十進数の秒数で指定します。<i>poll-interval</i> が 0 の場合は、ポリシーの check 関数は呼ばれません。</p> <p><i>policy-name</i> ファイル名。ポリシーを実行するファイルのパス名を記述します。デフォルトのポリシーファイルは以下のディレクトリに置かれます。</p> <p style="text-align: center;">/usr/platform/SUNW,UltraSPARC-III-Netractor/lib/envmond/sparcv9</p> <p><i>policy-args</i> 初期化時にポリシーに渡される空白文字で区切られた引数のリスト。引数の形式や数はポリシーに依存します。</p> <p>以下では envmond に実装されているポリシーについて説明します。</p>

fancpu ポリシー fancpu ポリシーは I2C スレーブデバイスを *poll-interval* 秒ごとに監視し、現在の CPU の温度とファントレーの状態を取得します。CPU の温度が警告温度に達した場合は、警告メッセージがシステムコンソールに表示され、`syslog.conf(4)` で指定されるシステムログファイルに記録されます。CPU の温度が停止温度に達した場合は、停止メッセージが `syslogd(1M)` によってシステムコンソールに表示され、システムは `shutdown(1M)` コマンドにより停止します。ファンの状態はシステム状態ボードの LED に反映され、ログメッセージがシステムログデーモン `syslogd` に送られます。

powersupply ポリシー

`powersupply` ポリシーは電源装置の状態に従って、システム状態ボードの LED を点灯させます。このポリシーは電源装置の故障時の割り込みイベントにも対応します。

scsb ポリシー

システムコントローラおよび状態ボード (`scsb`) ポリシーは第一に `cPCI` スロットの状態 LED を制御する `scsb` ドライバを構成します。`scsb_led_ctrl` のデフォルト設定は `false` で、`scsb` ドライバが `cPCI` スロットの LED を制御することを示します。`scsb_led_ctrl` が `true` に設定された場合は、スロット LED の更新はアプリケーションが行います。

使用例

例 1 - エントリの例

最初の例では `powersupply` 共有ライブラリを 60 秒ごとに起動します。次の例では `scsb` ポリシーが `cPCI` スロットの状態 LED を制御するように指定します。

```
60 powersupply.so
60 scsb.so scsb_led_ctrl=false
```

ファイル

```
/usr/platform/SUNW,UltraSPARC-IIi-Netract/
インストールディレクトリ
```

以下の関連パスはすべて上記のディレクトリ下にあります。

lib/envmond/sparcv9/envmond

環境デーモンの実行形式

lib/envmond/sparcv9/fancpu.so

CPU 温度とファンの回転速度制御ポリシー

lib/envmond/sparcv9/powersupply.so

電源装置モニターポリシー

関連項目

syslogd(1M), syslogd.conf(4)

第4章

デバイスおよび
ネットワークインタフェース

名前	nf — FDDI デバイスドライバ
形式	<pre>#include <sys/nf.h> #include <sys/dlpi.h></pre>
機能説明	<p>nf は、マルチスレッド化された、読み込みや複製が可能な STREAMS ハードウェアデバイスドライバです。このドライバでは、SBus カード内の DP83265A (BSI-2) FDDI コントローラにおいてコネクションレス型のデータリンクプロバイダインタフェース (dlpi(7P)) を使用することができます。このドライバで使用することができる FDDI カードの数には制限がありません。</p> <p>nf ドライバは、BSI-2、BMAC、PLAYER+ ハードウェアの基本的な機能を提供します。機能には、チップの初期化、フレームの転送と受信、マルチキャストとプロミスキュアスへの対応、エラーからの回復および報告が含まれます。</p> <p>複製された文字型特殊デバイスの <code>/dev/nf</code> は、システムにインストールされたすべての BSI-2 コントローラに接続するために使用します。</p>
nf と DLPI	<p>nf ドライバは、「スタイル 2」のデータリンクサービスプロバイダです。すべての M_PROTO 型および M_PCPROTO 型のメッセージは、DLPI 基本式として解釈されます。開いているストリームに特殊デバイス (ppa) を関連付けるには、ユーザーからの明示的な DL_ATTACH_REQ メッセージが必要となります。この ppa ID は、符号なしの長整数として解釈され、対応するデバイスインスタンス (ユニット) 番号を示します。ppa フィールドの値がシステムにおける有効なデバイスインスタンス番号に対応していない場合は、エラー (DL_ERROR_ACK) が nf ドライバによって返されます。デバイスは最初の接続時に初期化され、最後の切り離し時に終了 (停止) します。</p> <p>ユーザーからの DL_INFO_REQ に応答して nf ドライバが返す DL_INFO_ACK 基本式の値を以下に示します。</p> <ul style="list-style-type: none"> ■ SDU の最大値は 4352 (FDDIMTU) です。 ■ SDU の最小値は 0 です。 ■ dlsap アドレス長は 8 です。 ■ MAC 形式は DL_FDDI です。 ■ sap 長の値は -2 で、これは、DLSAP アドレスの中で物理アドレス構成要素のすぐ後に 2 バイト長の sap 構成要素が位置することを意味します。

- サービスモードは DL_CLDLS です。
- サービスの品質 (QOS: quality of service) オプションは現在のところ含まれていないため、QOS フィールドは 0 です。
- プロバイダスタイルは DL_STYLE2 です。
- バージョンは DL_VERSION_2 です。
- ブロードキャストアドレスの値は Ethernet/IEEE のブロードキャストアドレス (0xFFFFFFFF) です。

DL_ATTACHED 状態になると、特定の SAP (Service Access Pointer) をストリームと関連付けるために、ユーザーは DL_BIND_REQ を送る必要があります。nf ドライバは、DL_BIND_REQ 内の sap フィールドを Ethernet の「タイプ」として解釈します。したがって、sap フィールドの有効な値は [0-0xFFFF] の範囲になります。一度にストリームに接続できるのは 1 つのタイプの Ethernet だけです。

Ethernet V2 サービスに加えて、「802.3 モード」が nf ドライバによって提供され、次のように機能します。sap 値 0 は、ユーザーによる 802.3 モードの要求を示します。DL_BIND_REQ の sap フィールドの値が 0 の場合、nf ドライバは最初のメッセージの長さを M_PROTO mblk を含まずに計算し、MAC フレームヘッダーの長さフィールドにこの値を持ち、スナップヘッダーに 0xaa030000 の値を持つ 802.3 フレームを転送します。タイプフィールドが [0-1500] の範囲内にある媒体から受信したすべてのフレームは 802.3 フレームとみなされ、sap 値が 0 のすべての開いているストリームに配信されます。複数のストリームが 802.3 モードにある場合は、フレームは複製され、DL_UNITDATA_IND メッセージとして複数のストリームに配信されます。

nf ドライバの DLSAP アドレス形式は、6 バイトの物理 (FDDI) アドレス部分と、それに続く 2 バイトの sap (タイプ) 部分とで構成されます。アプリケーションでは、DLSAP アドレスの作成や削除を行う際に、実装に依存した DLSAP アドレス形式を明示的に使用せずに、DL_INFO_ACK 基本式で返される情報を使用してください。sap 長、完全な DLSAP 長、sap/物理の順序は DL_INFO_ACK に含まれます。物理アドレス長は、完全な DLSAP アドレス長から sap 長を引くか、DL_PHYS_ADDR_REQ を発行して、ストリームに対応する現在の物理アドレスを取得することによって計算することができます。

nf の基本式

DL_BOUND 状態になると、nf ドライバに DL_UNITDATA_REQ メッセージを送信することによって、ユーザーは FDDI 上でフレームを送信することができます。nf ドライバは、DL_UNITDATA_IND メッセージの FDDI のタイプと一致する sap 値を持ち、開いていて結合されているストリームすべてに、受信された FDDI フレームを配信します。受信された FDDI フレームは複製され、必要に応じて複数の開いているストリームに配信されます。

DL_UNITDATA_REQ および DL_UNITDATA_IND メッセージに含まれる DLSAP アドレスは、sap (タイプ) と物理 (FDDI) 部分で構成されます。

nf ドライバは、強制的なコネクションレス DLPI メッセージセットに加えて、以下の基本式に対応しています。

DL_ENABMULTI_REQ および DL_DISABMULTI_REQ 基本式は、個々のマルチキャストグループアドレスの受信を有効または無効にします。これらの基本式を使用して、一群のマルチキャストアドレスをストリームごとに繰り返し作成および変更することができます。これらの基本式は、DL_ATTACHED 後の状態にある場合に、nf ドライバによって受け付けられます。

dl_level フィールド内に DL_PROMISC_PHYS フラグが設定された状態での DL_PROMISCON_REQ および DL_PROMISCOFF_REQ の基本式は、ローカルホストによって生成されたフレームを含む、媒体上のすべての(「プロミスキュアモード」の)フレームの受信を有効または無効にします。

DL_PROMISC_SAP フラグが設定されている状態で使用すると、sap 値 (Ethernet のタイプ) の受信を有効または無効にします。

DL_PROMISC_MULTI フラグが設定されている状態で使用すると、すべてのマルチキャストグループアドレスの受信を有効または無効にします。どちらの効果も、常にストリームごとに適用され、当該ストリームやその他のストリームの sap および物理レベルの設定の影響は受けません。

DL_PHYS_ADDR_REQ 基本式は、DL_PHYS_ADDR_ACK 基本式の中のストリームに対応する (接続されている) MAC アドレス (6 桁の数) を返します。この基本式は、DL_ATTACH_REQ の成功後の状態においてのみ有効です。

DL_SET_PHYS_ADDR_REQ 基本式は、このストリームに対応する (接続されている) MAC アドレスを変更します。このストリームを最初に開く処理は、スーパーユーザーのみが行うことができます。スーパーユーザーでない場合は、DL_ERROR_ACK で EPERM が返されます。この基本式は、このデバイスに接続された現在および未来のすべてのストリームに影響する意味で破壊的です。この基本式がこのストリームに対して成功すると、このデバイスに接続

された他のすべてのストリームに **M_ERROR** が送信されます。物理アドレスが一度変更されると、その後開かれ、このデバイスに接続されたストリームは、この新しい物理アドレスを取得します。物理アドレスが一度変更されると、物理アドレスを再び変更するためにこの基本式が使用されるか、システムが再起動されるまでこの物理アドレスは存続します。

デフォルトでは、最初のインタフェースはシステムの **MAC** アドレスを使用しますが、後続のインタフェースは **FDDI** ローカルアドレスを使用します。

ファイル

/dev/nf

関連項目

smt(7), dlpi(7)

名前	pf — FDDI デバイスドライバ
形式	<pre>#include <sys/pf.h> #include <sys/dlpi.h></pre>
機能説明	<p>pf は、マルチスレッド化された、読み込みや複製が可能な STREAMS ハードウェアデバイスドライバです。このドライバでは、PCI カード内の PBS FDDI コントローラにおいてコネクションレス型のデータリンクプロバイダインタフェース (dlpi(7P)) を使用することができます。また、このドライバでは、ステーション管理の統計情報と状態情報を取得するためのアプリケーションを使用することができます。このドライバで使用することができる FDDI カードの数には制限がありません。pf ドライバは、PBS、BMAC、PLAYER+ ハードウェアの基本的な機能を提供します。機能には、チップの初期化、LLC/SMT フレームの転送と受信、マルチキャストとプロミスキューアスへの対応、エラーからの回復および報告が含まれます。</p> <p>複製された文字型特殊デバイスの <code>/dev/pf</code> は、システムにインストールされたすべての PBS コントローラに接続するために使用します。</p>
pf と DLPI	<p>pf ドライバは、「スタイル 2」のデータリンクサービスプロバイダです。すべての M_PROTO 型および M_PCPROTO 型のメッセージは、DLPI 基本式として解釈されます。開いているストリームに特殊デバイス (ppa) を関連付けるには、ユーザーからの明示的な DL_ATTACH_REQ メッセージが必要となります。この ppa ID は、符号なしの長整数として解釈され、対応するデバイスインスタンス (ユニット) 番号を示します。ppa フィールドの値がシステムにおける有効なデバイスインスタンス番号に対応していない場合は、エラー (DL_ERROR_ACK) が pf ドライバによって返されます。デバイスは最初の接続時に初期化され、最後の切り離し時に終了 (停止) します。</p> <p>ユーザーからの DL_INFO_REQ に応答して pf ドライバが返す DL_INFO_ACK 基本式の値を以下に示します。</p> <ul style="list-style-type: none"> ■ SDU の最大値は 4352 (FDDIMTU) です。 ■ SDU の最小値は 0 です。 ■ dlsap アドレス長は 8 です。 ■ MAC 形式は DL_FDDI です。

- sap 長の値は -2 で、これは、DLSAP アドレスの中で物理アドレス構成要素のすぐ後に 2 バイト長の sap 構成要素が位置することを意味します。
- サービスモードは DL_CLDLS です。
- サービスの品質 (QOS: quality of service) オプションは現在のところ含まれていないため、QOS フィールドは 0 です。
- プロバイダスタイルは DL_STYLE2 です。
- バージョンは DL_VERSION_2 です。
- ブロードキャストアドレスの値は Ethernet/IEEE のブロードキャストアドレス (0xFFFFFFFF) です。

DL_ATTACHED 状態になると、特定の SAP (Service Access Pointer) をストリームと関連付けるために、ユーザーは DL_BIND_REQ を送る必要があります。pf ドライバは、DL_BIND_REQ 内の sap フィールドを Ethernet の「タイプ」として解釈します。したがって、sap フィールドの有効な値は [0-0xFFFF] の範囲になります。一度にストリームに接続できるのは 1 つのタイプの Ethernet だけです。

Ethernet V2 サービスに加えて、「802.3 モード」が pf ドライバによって提供され、次のように機能します。sap 値 0 は、ユーザーによる 802.3 モードの要求を示します。DL_BIND_REQ の sap フィールドの値が 0 の場合、pf ドライバは最初のメッセージの長さを M_PROTO mblk を含まずに計算し、MAC フレームヘッダーの長さフィールドにこの値を持ち、スナップヘッダーに 0xaaaa030000 の値を持つ 802.3 フレームを転送します。タイプフィールドが [0-1500] の範囲内にある媒体から受信したすべてのフレームは 802.3 フレームとみなされ、sap 値が 0 のすべての開いているストリームに配信されます。複数のストリームが 802.3 モードにある場合は、フレームは複製され、DL_UNITDATA_IND メッセージとして複数のストリームに配信されます。

pf ドライバの DLSAP アドレス形式は、6 バイトの物理 (FDDI) アドレス部分と、それに続く 2 バイトの sap (タイプ) 部分とで構成されます。アプリケーションでは、DLSAP アドレスの作成や削除を行う際に、実装に依存した DLSAP アドレス形式を明示的に使用せずに、DL_INFO_ACK 基本式で返される情報を使用してください。sap 長、完全な DLSAP 長、sap/物理の順序は DL_INFO_ACK に含まれます。物理アドレス長は、完全な DLSAP アドレス

長から sap 長を引くか、DL_PHYS_ADDR_REQ を発行して、ストリームに対応する現在の物理アドレスを取得することによって計算することができます。

DL_BOUND 状態になると、pf ドライバに DL_UNITDATA_REQ メッセージを送信することによって、ユーザーは FDDI 上でフレームを送信することができます。pf ドライバは、DL_UNITDATA_IND メッセージの FDDI のタイプと一致する sap 値を持ち、開いていて結合されているストリームすべてに、受信された FDDI フレームを配信します。受信された FDDI フレームは複製され、必要に応じて複数の開いているストリームに配信されます。

DL_UNITDATA_REQ および DL_UNITDATA_IND メッセージに含まれる DLSAP アドレスは、sap (タイプ) と物理 (FDDI) 部分で構成されます。

pf の基本式

pf ドライバは、強制的なコネクションレス DLPI メッセージセットに加えて、以下の基本式に対応しています。

DL_ENABMULTI_REQ および DL_DISABMULTI_REQ 基本式は、個々のマルチキャストグループアドレスの受信を有効または無効にします。これらの基本式を使用して、一群のマルチキャストアドレスをストリームごとに繰り返し作成および変更することができます。これらの基本式は、DL_ATTACHED 後の状態にある場合に、pf ドライバによって受け付けられます。

dl_level フィールド内に DL_PROMISC_PHYS フラグが設定された状態での DL_PROMISCON_REQ および DL_PROMISCOFF_REQ の基本式は、ローカルホストによって生成されたフレームを含む、媒体上のすべての (「プロミスキュアモード」の) フレームの受信を有効または無効にします。

DL_PROMISC_SAP フラグが設定されている状態で使用すると、sap 値 (Ethernet のタイプ) の受信を有効または無効にします。

DL_PROMISC_MULTI フラグが設定されている状態で使用すると、すべてのマルチキャストグループアドレスの受信を有効または無効にします。どちらの効果も、常にストリームごとに適用され、当該ストリームやその他のストリームの sap および物理レベルの設定の影響は受けません。

DL_PHYS_ADDR_REQ 基本式は、DL_PHYS_ADDR_ACK 基本式の中のストリームに対応する (接続されている) MAC アドレス (6 桁の数) を返します。この基本式は、DL_ATTACH_REQ の成功後の状態においてのみ有効です。

pf と SMT

DL_SET_PHYS_ADDR_REQ 基本式は、このストリームに対応する (接続されている) **MAC** アドレスを変更します。このストリームを最初に開く処理は、スーパーユーザーのみが行うことができます。スーパーユーザーでない場合は、**DL_ERROR_ACK** で **EPERM** が返されます。この基本式は、このデバイスに接続された現在および未来のすべてのストリームに影響する意味で破壊的です。この基本式がこのストリームに対して成功すると、このデバイスに接続された他のすべてのストリームに **M_ERROR** が送信されます。物理アドレスが一度変更されると、その後開かれ、このデバイスに接続されたストリームは、この新しい物理アドレスを取得します。物理アドレスが一度変更されると、物理アドレスを再び変更するためにこの基本式が使用されるか、システムが再起動されるまでこの物理アドレスは存続します。

デフォルトでは、最初のインタフェースはシステムの **MAC** アドレスを使用しますが、後続のインタフェースは **FDDI** ローカルアドレスを使用します。

smt ドライバは、パケットのスルーput統計情報、再構成イベント、インタフェース例外情報を提供します。また、**PHY** に関する情報と **FORMAC** エラーカウンタを提供します。

ユーザーは、プログラム中の `#include <pfsmt.h>` の行の前に次の 2 行を追加する必要があります。

```
#define          SMT7_2          0
#define          CFG_YES         1
```

ドライバにアクセスするには、複製された文字型特殊デバイスの `/dev/pf` が使用されます。開かれたストリームを特定のデバイス (**ppa**) に関連付けるには、明示的に **DL_ATTACH_REQ** メッセージを使用する必要があります (**ppa** はインタフェースインスタンス番号に対応します)。

DL_ATTACHED 状態になると、**DL_BIND_REQ** を送信する必要はありません。 `ioctl(2)` 呼び出しを使用して、ドライバと対話することができます。 `ioctl` に対する引数は、以下の形式で指定します。

```
ioctl (int fd, int request, SMTCB *smtp)
```

request は、**smt** ドライバに固有であり、**SMT_GET** と **SMT_ACT** のいずれかを指定することができます。 **SMTCB** は、ヘッダーファイルの `pfsmt.h` に以下のように定義されています。

```
typedef struct {
    int          command;
    int          sub_command;
    int          param1;
    int          param2;
    int          param3;
    char         *where;
    int          length;
} SMTCB;
```

SMT_GET:

SMT_GET は、HPC レジスタを読み取ったり **smt** 状態を取得したりするためのさまざまな関数を提供します。**smt** の **command** フィールドは、以下のいずれかの値に初期化する必要があります。

```
HPC_BMAC1_REGS : BMAC レジスタの読み取り
HPC_READ       : HPC レジスタの読み取り
HPC_PORT1_REGS : RMT ポート 1 および
HPC_PORT2_REGS ポート 2 レジスタの読み取り
```

これらのコマンドには、サブコマンドが用意されているものがあります。

sub_command フィールドは、以下のいずれかのサブコマンドに初期化する必要があります。

1. HPC_BMAC1_REGS

HPC_BMAC1_REGS は、**BMAC** レジスタの読み取りを可能にします。**HPC_BMAC1_REGS** には、**GET_COUNTER_GROUP** および **GET_NEIGHBOR_ADDR** の 2 つのサブコマンドがあります。**GET_COUNTER_GROUP** は、さまざまな **SMT** カウンタ値を取得するために使用されます。

GET_COUNTER_GROUP を使用するには、**SMTCB *smt** を以下のように初期化する必要があります。

```
COUNTER_GROUP ct;

smt->command = HPC_BMAC1_REGS;
smt->sub_command = GET_COUNTER_GROUP;
smt->where = (char *) &ct;
smt->length = sizeof (ct);
```

GET_NEIGHBOR_ADDR は、隣接ステーションの MAC アドレスの取得を可能にします。GET_NEIGHBOR_ADDR を使用するには、SMTCB *smtp を以下のように初期化する必要があります。

```
char                addr_buf[12];

smtp->command = HPC_BMAC1_REGS;
smtp->sub_command = GET_NEIGHBOR_ADDR;
smtp->where = addr_buf; smtp->length = 12;
```

2. HPC_READ

HPC_READ は、HPC レジスタの読み取りを可能にします。HPC_READ には、サブコマンドはありません。HPC_READ を使用するには、SMTCB *smtp を以下のように初期化する必要があります。

```
smtp->command = HPC_READ;
smtp->param1 = HPC_READ | HPC_SIZE_BYTE
              | <;HPC_reg_offset>;
smtp->where = (char *) smtp;
```

HPC_register_offset オフセットは、HPC が提供するレジスタ空間です。レジスタのオフセットについては、pfsmt.h ファイルを参照してください。

3. HPC_PORT1_REGS と HPC_PORT2_REGS

HPC_PORT1_REGS は、接続管理の状態の取得を可能にします。HPC_PORT2_REGS は第 2 ポート用です (DAS インタフェースのみ)。HPC_PORT1_REGS には、GET_PORT_GROUP というサブコマンドがあります。HPC_PORT1_REGS を使用するには、SMTCB *smtp を以下のように初期化する必要があります。

```
FDDI_PORT_GROUP port;
smtp->command = HPC_PORT1_REGS;
smtp->sub_command = GET_PORT_GROUP;
smtp->where = (char *) &port;
smtp->length = sizeof (port);
```

port.ecm_state および port.pcm_state は、構造体の port に返される 2 つの重要な状態を表します。port.ecm_state は、ECM 状態マシンの現在の状態に対応しています。有効な値は、OUT、IN、TRACE、PATHTEST、INSERT、CHECK、DEINSERT です。port.ecm_state に

返される値は、ECM 状態のリストへのインデックスになります。
port.pcm_state は、PCM 状態マシンの現在の状態に対応しています。
 有効な値は、OFF、BREAK、TRACE、CONNECT、NEXT、
 SIGNAL、JOIN、VERIFY、ACTIVE、MAINT です。**port.pcm_state**
 に返される値は、PCM の状態のリストへのインデックスになりま
 す。

SMT_ACT:

SMT_ACT は、**smt** ドライバの状態を設定するためのコマンドです。
command フィールドには、必ず **SMT_CTL** を設定します。**SMT_ACT** には、
SMT_ACCEPT_FRAME と **SMT_CLOSE** の 2 つのサブコマンドがあります。
SMT_ACCEPT_FRAME は、SMT API クライアントが活動中のときに使用する
 必要があります。

```
smt->command = SMT_CTL;
smt->sub_command = SMT_ACCEPT_FRAME;
```

SMT_CLOSE は、API クライアントが終了するときに使用する必要がありま
 す。

```
smt->command = SMT_CTL;
smt->sub_command = SMT_CLOSE;
```

SMT NSA フレームを転送するには、ユーザーは **FDDI_NSA sap** に接続する
 必要があります。他の **SMT** フレームを転送するには、**FDDI SMTINFO sap** に
 結合することができます。

ファイル

/dev/pf

関連項目

dlpi(7)

名前	smt — FDDI SMT アプリケーションインタフェース用デバイスドライバ
形式	<code>#include <sys/nfsmt.h></code>
機能説明	<p>smt は、マルチスレッド化された、読み込みや複製が可能な STREAMS ハードウェアデバイスドライバです。このドライバでは、アプリケーションプログラムがステーション管理の統計情報と状態情報を取得するためのデータリンクプロバイダインタフェース dlpi(7) を使用することができます。smt ドライバは、パケットのスループット統計情報、再構成イベント、インタフェース例外情報を提供します。また、PHY に関する情報と FORMAC エラーカウンタを提供します。</p> <p>ユーザーは、プログラム中の <code>#include <nfsmt.h></code> の行の前に次の 2 行を追加する必要があります。</p> <pre> #define SMT7_2 0 #define CFG_YES 1 </pre> <p>ドライバにアクセスするには、複製された文字型特殊デバイスの <code>/dev/smt</code> が使用されます。開かれたストリームを特定のデバイス (ppa) に関連付けるには、明示的に DL_ATTACH_REQ メッセージを使用する必要があります (ppa はインタフェースインスタンス番号に対応します)。</p> <p>DL_ATTACHED 状態になると、DL_BIND_REQ を送信する必要はありません。ioctl(2) 呼び出しを使用して、ドライバと対話することができます。ioctl に対する引数は、以下の形式で指定します。</p> <pre> ioctl (int fd, int request, SMTCB *smtcb) </pre> <p>request は、smt ドライバに固有であり、SMT_GET と SMT_ACT のいずれかを指定することができます。SMTCB は、ヘッダーファイルの <code>nfsmt.h</code> に以下のように定義されています。</p> <pre> typedef struct { int command; int sub_command; int param1;int param2; int param3; char *where; int length; }SMTCB; </pre>

SMT_GET:

SMT_GET は、HPC レジスタを読み取ったり **smt** 状態を取得したりするためのさまざまな関数を提供します。**smt** の **command** フィールドは、以下のいずれかの値に初期化する必要があります。

```

HPC_BMAC1_REGS : BMAC レジスタの読み取り
HPC_READ       : HPC レジスタの読み取り
HPC_PORT1_REGS : RMT ポート 1 および
HPC_PORT2_REGS   ポート 2 レジスタの読み取り

```

これらのコマンドには、サブコマンドが用意されているものがあります。**sub_command** フィールドは、以下のいずれかのサブコマンドに初期化する必要があります。

1. HPC_BMAC1_REGS

HPC_BMAC1_REGS は、BMAC レジスタの読み取りを可能にします。**HPC_BMAC1_REGS** には、**GET_COUNTER_GROUP** および **GET_NEIGHBOR_ADDR** の 2 つのサブコマンドがあります。**GET_COUNTER_GROUP** は、さまざまな SMT カウンタ値を取得するために使用されます。

GET_COUNTER_GROUP を使用するには、**SMTCB *smt** を以下のように初期化する必要があります。

```

COUNTER_GROUP ct;
smt->command = HPC_BMAC1_REGS;
smt->sub_command = GET_COUNTER_GROUP;
smt->where = (char *) &ct;
smt->length = sizeof (ct);

```

GET_NEIGHBOR_ADDR は、隣接ステーションの MAC アドレスの取得を可能にします。**GET_NEIGHBOR_ADDR** を使用するには、**SMTCB *smt** を以下のように初期化する必要があります。

```

char addr_buf[12];
smt->command = HPC_BMAC1_REGS;
smt->sub_command = GET_NEIGHBOR_ADDR;
smt->where = addr_buf;
smt->length = 12;

```


2. HPC_READ

HPC_READ は、HPC レジスタの読み取りを可能にします。

HPC_READ には、サブコマンドはありません。HPC_READ を使用するには、SMTCB *smtp を以下のように初期化する必要があります。

```
smtp->command = HPC_READ;
smtp->param1 = HPC_READ | HPC_SIZE_BYTE
              |<HPC_reg_offset>;
smtp->where = (char *) smtp;
```

HPC_register_offset オフセットは、HPC が提供するレジスタ空間です。レジスタのオフセットについては、nfsmt.h ファイルを参照してください。

3. HPC_PORT1_REGS と HPC_PORT2_REGS

HPC_PORT1_REGS は、接続管理の状態の取得を可能にします。

HPC_PORT2_REGS は第 2 ポート用です (DAS インタフェースのみ)。HPC_PORT1_REGS には、GET_PORT_GROUP というサブコマンドがあります。HPC_PORT1_REGS を使用するには、SMTCB *smtp を以下のように初期化する必要があります。

```
FDDI_PORT_GROUP port;
smtp->command = HPC_PORT1_REGS;
smtp->sub_command = GET_PORT_GROUP;
smtp->where = (char *) &port;
smtp->length = sizeof (port);
```

port.ecm_state および port.pcm_state は、構造体の port に返される 2 つの重要な状態を表します。port.ecm_state は、ECM 状態マシンの現在の状態に対応しています。有効な値は、OUT、IN、TRACE、PATHTEST、INSERT、CHECK、DEINSERT です。port.ecm_state に返される値は、ECM 状態のリストへのインデックスになります。

port.pcm_state は、PCM 状態マシンの現在の状態に対応しています。有効な値は、OFF、BREAK、TRACE、CONNECT、NEXT、SIGNAL、JOIN、VERIFY、ACTIVE、MAINT です。port.pcm_state に返される値は、PCM の状態のリストへのインデックスになります。

SMT_ACT:

SMT_ACT は、smt ドライバの状態を設定するためのコマンドです。
command フィールドには、必ず SMT_CTL を設定します。SMT_ACT には、SMT_ACCEPT_FRAME と SMT_CLOSE の 2 つのサブコマンドがあります。SMT_ACCEPT_FRAME は、SMT API クライアントが活動中のときに使用する必要があります。

```
smt->command = SMT_CTL;  
smt->sub_command = SMT_ACCEPT_FRAME;
```

SMT_CLOSE は、API クライアントが終了するときに使用する必要があります。

```
smt->command = SMT_CTL;  
smt->sub_command = SMT_CLOSE;
```

ファイル

/dev/smt

関連項目

nf(7), dlpi(7)

第5章

デバイス

名前	ge — GEM Gigabit-Ethernet デバイスドライバ
形式	/dev/ge
機能説明	<p>ge GigabitEthernet ドライバ (以下 ge ドライバと略記) は、マルチスレッド化され、読み込みや複製が可能な STREAMS ハードウェアドライバです。GEM SBus および GEM PCI GigabitEthernet アドイン型アダプタに対してコネクシヨンレス型のデータリンクプロバイダインタフェース (DLPI: dlpi(7P) をサポートします。ge ドライバは、システムに組み込まれた、複数の GEM アダプタをサポートするものです。ge ドライバは主に GEM 対応の Ethernet ハードウェアをサポートします。また、SUNW, sbus-gem (SBus 対応 GEM) や pci108e, 2bad (PCI 対応 GEM) などのデバイスを制御します。その機能は、チップの初期化、フレームの転送と受信、マルチキャストとプロミスキュアスのサポート、エラーの復旧処理および報告です。GEM デバイスは、外部 SERDES および光ファイバー通信トランシーバを制御する GEM ASIC (専用チップ) を用いて、100BASE-SX のネットワークインタフェースを提供します。GEM ASIC は、適切なバスインタフェースに対し、MAC 機能や PCS (Physical Code Sub-layer: 物理コードサブレイヤー) 機能を提供します。外部 SERDES は、光ファイバートランシーバに接続され、物理的な接続を提供します。</p> <p>1000BASE-SX 規格には、自動的に動作モードを選択する「自動ネゴシエーション」機能が規定されています。さらに全二重モードのために、GEM ASIC は、IEEE 802.3x フロー制御に準拠した自動ネゴシエーション機能を実現しています。GEM PCS 機能は、遠隔リンク先 (リンクパートナー) に対して、「自動ネゴシエーション」を実行したり、遠隔リンク先からのメッセージを受信したりします。また、優先度に応じて最も優先順位の高いモードを選択します。ドライバが対応しているシステムでは、強制モードもサポートします。</p>
アプリケーション プログラミング インタフェース	<p>文字複製された特殊デバイスである /dev/ge は、システムに組み込まれている、すべての ge コントローラへのアクセスに使用されます。</p>
ge と DLPI	<p>ge ドライバは、「スタイル 2」型のデータリンクサービスを提供するものです。すべての M_PROTO 型、および M_PCPROTO 型のメッセージは、DLPI プリミティブとして解釈されます。DLPI プリミティブについては、<sys/dlpi.h> ファイルで定義されています。詳細な情報については、dlpi(7P) に関するマニュアルページを参照してください。ユーザーからの DL_ATTACH_REQ メッセージは、特殊デバイス (ppa) に対して開かれたストリームに関連付けられなければなりません。ppa ID は、unsigned long 型</p>

のデータとして解釈され、対応するデバイスインスタンス (ユニット) 番号を指し示します。ppa フィールドの値が、そのシステムの有効なデバイスインスタンス番号と一致しない場合、ドライバは、エラー値として (DL_ERROR_ACK) を返します。デバイスは、最初の接続時に初期化され、最後の接続時に終了処理 (停止) されます。

ユーザーからの DL_INFO_REQ に対して ge ドライバによって返される DL_INFO_ACK プリミティブの値は、以下のようになります。

- SDU の最大値は、1500 です (ETHERMTU <sys/ethernet.h> で定義されている)。
- SDU の最小値は、0 です。
- dlsap のアドレス長は、8 です。
- MAC 型は、DL_ETHER です。
- sap 長の値は、-2 です。これは、物理アドレスコンポーネントの直後に、DLSAP アドレス内部にある、2 バイトの sap コンポーネントが続くことを意味しています。
- サービスモードは、DL_CLDLS です。
- オプションなしのサービス品質 (QOS) をサポートする場合、QOS フィールドの値は、0 になります。
- サービス提供の形式は、DL_STYLE2 です。
- バージョンは、DL_VERSION_2 です。
- ブロードキャストアドレスの値は、Ethernet/IEEE ブロードキャストアドレスです (0xFFFFFFFF)。

DL_ATTACHED 状態において特殊な SAP (Service Access Pointer) をストリームに関連付けるために、ユーザーは DL_BIND_REQ を送信する必要があります。sap フィールドの値が [0-0xFFFF] の範囲内にある限り、ge ドライバは、DL_BIND_REQ 中の sap フィールドを、Ethernet の“種類”として解釈します。いかなる場合も、ただ 1 つの種類の Ethernet をストリームに結合することができます。

ユーザーが `sap` の値として 0 を指定した場合、受け取り側は、“802.3 モード” になります。“`type`” フィールドの値が [0-1500] の範囲内にあるメディアを受信したすべてのフレームは、802.3 準拠とみなされます。そして、`sap` 値が 0 に結合されている、開いているすべてのストリームに経路付けられます。1 つ以上のストリームが “802.3 モード” であるなら、フレームは複製され、`DL_UNITDATA_IND` メッセージとして、複数のストリームに経路付けられます。

データ伝送中に、`sap` 値が 0 で、なおかつ `destination type` フィールドの値が [0-1500] の範囲内であれば、`ge` ドライバは、`DL_BIND_REQ` 中の `sap` フィールドをチェックします。もし、いずれの条件も真の場合、デバイスドライバは、すべての `DL_UNITDATA_REQ` メッセージの直後に続く最初の `M_PROTO mblk` (メッセージブロック) を含めずに、メッセージの長さを算出します。そして、この値を `MAC` フレームヘッダ長フィールドに持つ 802.3 フレームを送信します。

`ge` ドライバの `DLSAP` アドレスの書式は、8 バイトの `DLSAP` アドレスを生成する、2 バイトの `sap` (型) コンポーネントと、それに続く (Ethernet の) 6 バイトの物理アドレスコンポーネントから成り立っています。アプリケーションは、この特殊な実装を指定する `DLSAP` アドレスの書式をハードコードすべきではありません。`DLSAP` アドレスの構成と分解を行う `DL_INFO_ACK` プリミティブに返される情報を使用すべきです。`sap` のデータ長、`DLSAP` の全データ長、`sap` または物理的な指示は、`DL_INFO_ACK` の中に含まれています。物理アドレス長は、全 `DLSAP` アドレス長から `sap` のデータ長を引くか、ストリームに関連付けられた現在の物理アドレスを得る、`DL_PHYS_ADDR_REQ` を発行することにより、算出することができます。

`DL_BOUND` ステートでは、`ge` ドライバに `DL_UNITDATA_REQ` メッセージを送ることによって、ユーザーは Ethernet 上にフレームを送信することができます。`ge` ドライバは、`DL_UNITDATA_IND` メッセージに Ethernet タイプが一致している `sap` を持つ、開かれた、もしくは結合されたストリームに、受信した Ethernet フレームを経路付けすることになります。必要に応じて、受信された Ethernet フレームは複製され、複数の開かれたストリームに経路付けされます。`DL_UNITDATA_REQ` および `DL_UNITDATA_IND` メッセージの中に含まれる `DLSAP` アドレスは、`sap` (型) と物理 (Ethernet) コンポーネントから構成されています。

ge プリミティブ

強制コネクションレス DLPI メッセージセットに加えて、さらにge ドライバは、以下のプリミティブをサポートしています。

DL_ENABMULTI_REQ と **DL_DISABMULTI_REQ** プリミティブは、個々のマルチキャストグループアドレスの受け取りを許可・不許可します。この 2 つのプリミティブを使うことによって、マルチキャストアドレスの組み合わせが各ストリーム上で何度も生成されたり、変更されたりします。ドライバは、**DL_ATTACHED** に続くいかなるステートにおいてもこの 2 つのプリミティブを受け入れます。

dl_level 中の **DL_PROMISC_PHYS** フラグセットとともに、**DL_PROMISCON_REQ** と **DL_PROMISCOFF_REQ** プリミティブは、ローカルホストの生成したフレームを含むメディアにある、すべてのフレーム (“プロミスキュアモード”) の受け入れを許可・不許可します。

DL_PROMISC_SAP フラグセットとともに使用されるとき、この機能は、すべての **sap** (**Ethernet** の種類を表す) の値の受け入れを許可・不許可します。

DL_PROMISC_MULTI フラグセットとともに使用されるとき、この機能は、すべてのマルチキャストグループアドレスの受け入れを許可・不許可します。それぞれの効果は、常にそれぞれのストリームの土台となり、該当するストリーム、または他のストリームの、他の **sap** および物理レベルの設定に依存します。

DL_PHYS_ADDR_REQ プリミティブは、**DL_PHYS_ADDR_ACK** プリミティブ中のストリームにその時点で関連 (結合) 付けられた 6 オクテットの **Ethernet** アドレスを返します。このプリミティブは、成功した **DL_ATTACH_REQ** の後に続く状態であるときのみ有効です。

DL_SET_PHYS_ADDR_REQ プリミティブは、このストリームにその時点で関連 (結合) 付けられた 6 オクテットの **Ethernet** アドレスを変更します。最初にこのストリームを開いたプロセスの身元保証 (**credentials**) はスーパーユーザでなければなりません。そうでない場合は、**EPERM** は、**DL_ERROR_ACK** 中に戻されます。このプリミティブは、このデバイスに結合された、現在および未来の他のすべてのストリームに影響を及ぼす点で、破壊的なものです。このプリミティブがこのストリーム上で成功したとき、このデバイスに結合された他のすべてのストリームに **M_ERROR** が送られます。一旦変更があると、その後に関かれ、このデバイスに結合されたすべてのストリームは、この新しい物理アドレスを取得するはずで、変更があったとき、この物理アドレスは、さらに次の変更があるか、システムが再起動されるまで保持されます。

ge ドライバ

デフォルトでは、リンクのモードとフロー制御機能を選択するために、ge ドライバは、“自動ネゴシエーション”を実行します。

リンクは、以下に示す 4 つのモードのいずれかになります。

- 1000 Mbps、全二重接続
- 1000 Mbps、半二重接続
- 対称的中断 (Symmetric Pause)
- 非対称的中断 (Asymmetric Pause)

上記の通信速度とモードは、1000BASE-TX 規格に規定されています。

自動ネゴシエーションプロトコルは、以下の項目を自動的に選択します。

- 動作モード (半二重または全二重)
- フロー制御機能 (対称または非対称、あるいはその両方)

自動ネゴシエーションプロトコルは、以下の処理を行います。

- リンク先パートナーのサポートする全動作モードの取得
- リンク先パートナーに対する、各機能を通知
- 優先度に基づいた、最も優先度の高いモードの選択

GEM ハードウェアは、上記の全動作モードを実現しています。デフォルト時は、リンク先との接続のために自動ネゴシエーションを使用し、次に、リンク先パートナーと同じ動作モードを選択します。**PCS** は、強制モードもサポートしています。これは、`ndd` ユーティリティを使用して、ドライバが動作モードとフロー制御機能を選択できるモードです。

GEM デバイスは、プログラム可能な IPG (Inter-Packet Gap) パラメタの `ipg1` および `ipg2` をサポートしています。デフォルトでは、ドライバは、`ipg1` を 8 バイトの倍数に、`ipg2` を 4 バイトの倍数に設定します (これらは標準的な数値です)。ユーザーは、これらの数値を標準的な 1000 Mbps IPG セットから 0.096 マイクロ秒へ変更してもかまいません。

デバイス

ge(7D)

ge のパラメタ リスト

ge ドライバは、GEM デバイスの各種パラメタを設定したり、取得することができます。パラメタのリストには、現在のトランシーバの状態、現在の接続の状態、パケット間隔、PCS の各機能、リンク先パートナーの各機能が含まれています。

PCS には、機能別に 2 つのセットがあります。1 つは、ハードウェアの各機能を反映したものであり、読み取り専用 (RO) パラメタです。2 つめのセットは、ユーザーによって選択された数値を反映したもので、高速選択 (speed selection) として使用されます。こちらは、読み取り・書き込み (RW) いずれも可能です。ブート時には、これらの 2 つの機能セットは同等です。リンク先パートナーの各機能もまた、読み取り専用パラメタです。なぜなら、そのときのデフォルト値は、読み取りのみが可能で、変更できないからです。

ファイル

/dev/ge ge 文字型特殊デバイス

/kernel/drv/ge.conf システム全般のデフォルトデバイスドライバ
属性

関連項目

ndd(1M), netstat(1M), driver.conf(4), dlpi(7P), ie(7D), le(7D)
hme(7D), qfe(7D)

ge(7D)

デバイス

索引

C

cdrw(1), 2
cfgadm_sbd(1M), 10

E

envmond(1M), 27
envmond.conf(4), 80

G

ge(7D), 100

N

nf(7), 84
nf_fddidaemon(1M), 29
nf_install_agents(1M), 30
nf_macid(1M), 31
nf_smtmon(1M), 32
nf_snmd(1M), 34
nf_snmd_kill(1M), 36
nf_stat(1M), 37
nf_sync(1M), 41

P

pf(7), 88
pf_fddidaemon(1M), 42
pf_install_agents(1M), 43
pf_macid(1M), 44
pf_smtmon(1M), 45
pf_snmd(1M), 47
pf_snmd_kill(1M), 49
pf_stat(1M), 50

R

rscadm(1M), 54

S

smt(7), 95
sunvts(1M), 59

V

vts_cmd(1M), 61
vtsk(1M), 68
vtsprobe(1M), 70
vtstty(1M), 73
vtsui(1M), 76
vtsui.ol(1M), 77

て

デバイス, 99

デバイスおよびネットワークインタフェース, 83

ふ

ファイル形式, 79

ほ

保守コマンド, 9

ゆ

ユーザーコマンド, 1