



Sun Enterprise™ 10000 SSP 3.5 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. 816-2249-10
Revision A, 2001 年 11 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コーペイ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、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：	Sun Enterprise 10000 SSP 3.5 Reference Manual Part No: 806-7614-10 Revision A
-----	---



Sun Enterprise 10000 SSP の帰属

本ソフトウェアの著作権は、カリフォルニア大学、米国サン・マイクロシステムズ、その他、関係する個人または組織が所有します。個別ファイルに権利の放棄が明示されていない限り、本ソフトウェアに関係するあらゆるファイルには、下記条件が適用されます。

作者は、既存の著作権告知文があらゆるコピーに留められること、また告知文がそのまま頒布版に含まれることを条件に、いかなる目的でも本ソフトウェアおよび関連文書を使用、複製、変更、修正、頒布、ライセンスすることを許可します。この条件を満たす限り、使用にあたり、書面による合意、ライセンスの付与、使用料の支払いは必要ありません。本ソフトウェアに加えられた修正部分の著作権は、その適用を受ける各ファイルの先頭ページに新しい条件を明記する限り、その作者が所有するものとし、ここに規定されているライセンス条件に従う必要はありません。

作者が次に記す損害の可能性について事前の通知を受けていたとしても、作者および頒布元は、本ソフトウェア、その関連文書、またその派生物を使用することによって生じた直接、間接、特別、付随、結果損害についていかなる個人または組織にも責任を負いません。

作者および頒布元は、商品性、特定の目的への適合性、侵害行為がないことの黙示の保証を含めて、いかなる保証も行いません。本ソフトウェアは「現状のままのもの」として提供されており、作者および頒布元は、保守、サポート、アップデート、機能強化、修正を提供する義務を負いません。

米国政府関連の方は以下をお読みください。 Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

本ソフトウェア、scotty は、TCP/IP ネットワークに関する情報を取得するための、いくつかの特殊なコマンドからなる、簡単な tcl インタプリタです。 Copyright (c) 1993, 1994, 1995, J. Schoenwaelder, TU Braunschweig, Germany, Institute for Operating Systems and Computer Networks. この著作権告知文があらゆるコピーに付記されることを条件に、いかなる目的でも、無料で本ソフトウェアおよびその関連文書を使用、複製、修正、頒布することを許可します。 Braunschweig 大学は、本ソフトウェアの目的適合性についていかなる表明もしません。本ソフトウェアは、黙示および明示的な保証無しに「現状のまま」で提供されます。

目次

Intro(1M)	2
autoconfig(1M)	5
board_id(1M)	8
bringup(1M)	10
cancelcmdsync(1M)	14
cb_prom(1M)	18
cb_reset(1M)	19
cbs(1M)	20
check_host(1M)	21
cmdsync(1M)	22
datasyncd(1M)	26
domain_create(1M)	28
domain_history(1M)	31
domain_link(1M)	33
domain_remove(1M)	35
domain_rename(1M)	37
domain_status(1M)	39

domain_switch(1M)	41
domain_unlink(1M)	42
edd(1M)	44
edd_cmd(1M)	46
fad(1M)	48
fan(1M)	49
fod(1M)	52
hostinfo(1M)	53
hostint(1M)	54
hostreset(1M)	56
hostview(1M)	57
hpost(1M)	58
initcmdsycn(1M)	67
machine_server(1M)	71
netcon(1M)	72
netcon_server(1M)	76
netcontool(1M)	77
obp_helper(1M)	79
power(1M)	81
redx(1M)	89
runcmdsync(1M)	91
savecmdsync(1M)	92
setdatasync(1M)	96
setfailover(1M)	100
showcmdsync(1M)	103

showdatasync(1M)	105
showdevices(1M)	108
showfailover(1M)	115
showusage(1M)	117
sigbcmd(1M)	120
snmpd(1M)	123
ssp_backup(1M)	126
ssp_config(1M)	127
ssp_downgrade(1M)	129
ssp_restore(1M)	130
ssp_startup(1M)	131
ssp_unconfig(1M)	133
straps(1M)	134
sys_clock(1M)	135
sys_id(1M)	138
thermcal(1M)	141
thermcal_config(1M)	142
Intro(4)	144
blacklist(4)	145
cb_config(4)	148
cb_port(4)	150
domain_config(4)	151
domain_history(4)	153
edd.emc(4)	154
edd.erc(4)	160

fad_files(4) 170
postrc(4) 172
redlist(4) 206
ssp_resource(4) 209
ssp_to_domain_hosts(4) 210

第1章

保守コマンド

名前	Intro - Sun Enterprise 10000 SSP 3.5 管理																																				
機能説明	<p>この節では、Enterprise 10000 SSP 環境で実行可能なコマンド、スクリプト、およびプログラムについて説明します。Dynamic Reconfiguration(DR) コマンドの一覧に関しては、Intro-dr(1M) コマンドを参照してください。</p> <hr/> <p>注 - InterDomain Networks (IDN) 機能は、Solaris 7 オペレーティング環境の 11/99 リリースにおいて、初めて Solaris リリースでサポートされました。</p> <hr/>																																				
コマンド一覧	<p>以下のコマンドがサポートされます。</p> <table border="0"> <tr> <td>autoconfig</td> <td>scantool/ インターコネクトの自動設定</td> </tr> <tr> <td>board_id</td> <td>ボード ID の読み取りまたは書き込み</td> </tr> <tr> <td>bringup</td> <td>ドメインの設定と起動</td> </tr> <tr> <td>cancelcmdsycn</td> <td>コマンド同期コマンド</td> </tr> <tr> <td>cb_prom</td> <td>PROM のアップグレード</td> </tr> <tr> <td>cb_reset</td> <td>ハングした制御ボードのリセットと再起動</td> </tr> <tr> <td>cbs</td> <td>制御ボードサーバー</td> </tr> <tr> <td>check_host</td> <td>ドメインが立ち上がっているかどうかの判別</td> </tr> <tr> <td>cmdsycn</td> <td>コマンド同期コマンド</td> </tr> <tr> <td>datasyncd</td> <td>データ同期デーモン</td> </tr> <tr> <td>domain_create</td> <td>新規のドメインの作成または既存のドメインの再作成</td> </tr> <tr> <td>domain_history</td> <td>ドメインの履歴の表示または削除</td> </tr> <tr> <td>domain_link</td> <td>ドメインのリンクによる InterDomain Networks (IDN) の形成・拡張</td> </tr> <tr> <td>domain_remove</td> <td>既存のドメインの削除</td> </tr> <tr> <td>domain_rename</td> <td>既存のドメインの名前変更</td> </tr> <tr> <td>domain_status</td> <td>ドメインの状態の表示</td> </tr> <tr> <td>domain_switch</td> <td>SUNW_HOSTNAME に指定されているドメイン名の変更</td> </tr> <tr> <td>domain_unlink</td> <td>InterDomain Networks (IDN) からのドメインのリンク解除</td> </tr> </table>	autoconfig	scantool/ インターコネクトの自動設定	board_id	ボード ID の読み取りまたは書き込み	bringup	ドメインの設定と起動	cancelcmdsycn	コマンド同期コマンド	cb_prom	PROM のアップグレード	cb_reset	ハングした制御ボードのリセットと再起動	cbs	制御ボードサーバー	check_host	ドメインが立ち上がっているかどうかの判別	cmdsycn	コマンド同期コマンド	datasyncd	データ同期デーモン	domain_create	新規のドメインの作成または既存のドメインの再作成	domain_history	ドメインの履歴の表示または削除	domain_link	ドメインのリンクによる InterDomain Networks (IDN) の形成・拡張	domain_remove	既存のドメインの削除	domain_rename	既存のドメインの名前変更	domain_status	ドメインの状態の表示	domain_switch	SUNW_HOSTNAME に指定されているドメイン名の変更	domain_unlink	InterDomain Networks (IDN) からのドメインのリンク解除
autoconfig	scantool/ インターコネクトの自動設定																																				
board_id	ボード ID の読み取りまたは書き込み																																				
bringup	ドメインの設定と起動																																				
cancelcmdsycn	コマンド同期コマンド																																				
cb_prom	PROM のアップグレード																																				
cb_reset	ハングした制御ボードのリセットと再起動																																				
cbs	制御ボードサーバー																																				
check_host	ドメインが立ち上がっているかどうかの判別																																				
cmdsycn	コマンド同期コマンド																																				
datasyncd	データ同期デーモン																																				
domain_create	新規のドメインの作成または既存のドメインの再作成																																				
domain_history	ドメインの履歴の表示または削除																																				
domain_link	ドメインのリンクによる InterDomain Networks (IDN) の形成・拡張																																				
domain_remove	既存のドメインの削除																																				
domain_rename	既存のドメインの名前変更																																				
domain_status	ドメインの状態の表示																																				
domain_switch	SUNW_HOSTNAME に指定されているドメイン名の変更																																				
domain_unlink	InterDomain Networks (IDN) からのドメインのリンク解除																																				

edd	イベント検出デーモン
edd_cmd	edd へのコマンドの送信
fad	ファイルアクセスデーモン
fan	ファンの電源と回転速度の表示または制御
fod	フェイルオーバーデーモン
hostinfo	システム情報の表示
hostint	プロセッサへの割り込みと、カーネルのコアの ダンプ
hostreset	ハングしたドメインのリセット
hostview	システムを監視するためのグラフィカルユー ザーインタフェース (GUI)
hpost	JTAG 経由の POST の制御と処理
initcmdsync	コマンド同期コマンド
machine_server	マシンサーバー
netcon	ネットワークコンソール
netcon_server	ネットワークコンソールのサーバーデーモン
netcontool	ネットワークコンソールのツール
obp_helper	システムメモリーへの OpenBoot のダウンロード
power	電源の制御
redx	リモートエミュレーションデバッガ
runcmdsync	フェイルオーバー後の回復のための指定された スクリプトの準備
savecmdsync	コマンド同期コマンド
setdatasync	データの同期で使用するデータ伝播リストの変 更
setfailover	SSP または制御ボードに対する自動フェイル オーバー状態の変更
showcmdsync	現在のコマンド同期リストを表示
showdatasync	フェイルオーバーの起動前に SSP のデータ同期 状態を表示
showdevices	システムボードのデバイスとリソースの使用状 況に関する情報の表示

showfailover	SSP と制御ボードのフェイルオーバー状態を表示
showusage	DR およびボードリソース情報の表示
sigbcmd	SSP からドメインへのコマンドの送信
snmpd	Sun Enterprise 10000 プラットフォームの SNMP プロキシエージェント
ssp_backup	SSP 環境のバックアップ
ssp_config	SSP または制御ボードの設定
ssp_downgrade	以前のバージョンの SSP に戻す (ダウングレード) よう SSP および Sun Enterprise 10000 システムを準備する
ssp_restore	SSP 環境の復元
ssp_startup	SSP デーモンの起動
ssp_unconfig	SSP の設定の解除
straps	SNMP トラップシンクサーバー
sys_clock	システムクロック周波数の表示・変更・設定
sys_id	SSP の IDPROM に設定されたシステム ID の表示または変更
thermcal	サーミスターのキャリブレーションの読み取りまたは書き込み
thermcal_config	SSP サーミスターキャリブレーションのデータファイルの作成

名前	autoconfig - scantool/ インターコネクトの自動設定
形式	autoconfig [<i>board_name</i>]
注意	このコマンドは、購入先の技術者のみが使用します。このコマンドを実行するには、ユーザー <code>ssp</code> である必要があります。
機能説明	<p>autoconfig は、Sun Enterprise 10000 システムに装着されているすべての設定済みのシステムボードと、2 つのセンタープレーン (センタープレーンハーフ) のチップのバージョンを順番にチェックします。次に、ファイル <code>\$SSPVAR/data/Ultra-Enterprise-10000/common/board/sysboard/chip.ids</code> (システムボードの場合)、またはファイル <code>\$SSPVAR/data/Ultra-Enterprise-10000/common/board/centerplane/chip.ids</code> (センタープレーンハーフの場合) を調べて、選択されたボードのシグニチャーを判別します。さらに、同じディレクトリ内のファイル <code>board.sigs</code> を調べ、選択されたシステムボードまたはセンタープレーンハーフを Scantool データベース内で更新するかどうか、および更新する場合にはその方法を判別します。</p> <p>注意： 現在オペレーティングシステムを実行中のシステムボードに対してこのコマンドを実行しないでください。また、オペレーティングシステムを実行中のドメインがある場合には、センタープレーンに対してこのコマンドを実行しないでください。</p> <p>autoconfig により以下の処理が実行されます。</p> <ol style="list-style-type: none"> 1. 選択されたボードのすべてのリングからチップ ID を読み取り、リング上のチップの ID 値および数を判別します。 2. チップ ID がすべて有効であることを確認し、可変長リングのモジュールタイプを判別します。 3. 選択されたボード上の各リングについて、リングシグニチャーを作成します。 4. <code>board.sigs</code> ファイルの情報から Scantool データベースで更新する必要があるリングを判別し、<code>board.sigs</code> ファイルに新しいリングシグニチャーを追加します。 5. Scantool データベースで更新する必要があるすべての可変長リングに適切なリングテンプレートを選択します。 6. 必要に応じて、ボードの新しいバージョンを作成することにより Scantool データベースを更新します。続いて、<code>board.sigs</code> ファイルに新しいバージョンのシグニチャーを追加します。

7. このボードの新しいバージョンを使用して、`$$SSPVAR/data/Ultra-Enterprise-10000/config/platform_name.config` を更新します。

リング上にエラーを検出した場合は、`autoconfig` はそのリングを変更せずに、処理を続行します。`chip.ids` ファイルまたは `board.sigs` ファイルのアクセス時、あるいは `Scantool` データベースの編集時にエラーが発生した場合は、`autoconfig` は選択されたボード用のファイルを復元してから、処理を続行します。

`autoconfig` の処理が完了すると、`SSP` を再起動して新しい `Scantool` データベースの起動を促すプロンプトが表示されます。再起動する場合は、シェルの環境変数が正しく設定されるように、ユーザー `ssp` でログインしていることを確認してください。

オプション

以下のオプションがサポートされています。

`board_name` 指定されたボードを設定します。有効なボード名は `sysboard n` (n は 0 ~ 15 の整数) または `centerplane n` (n は 0 または 1) です。このオプションを使用できるのは、システムボードを別のスロットへ移動したとき、新しいシステムボードを追加したとき、あるいはセンタープレーンを交換したときに限られます。

ファイル

`$$SSPVAR/data/Ultra-Enterprise/` ディレクトリ内の以下のファイルがサポートされます。

`/common/board/sysboard/chip.ds`

システムボード上のすべてのチップの ID が格納されます。

`/common/board/centerplane/chip.ids`

センタープレーン上のすべてのチップの ID が格納されます。

`/common/board/sysboard/board.sigs`

システムボードに対して作成されているすべてのリングシグニチャーおよびボードのバージョンシグニチャーが格納されます。

`/common/board/centerplane/board.sigs`

センタープレーンに対して作成されているすべてのリングシグニチャーとボードのバージョンシグニチャーが格納されます。

`/config/platform_name.config`

各システムボードとセンタープレーンの現在のバージョン番号が格納されます。

```
/common/board/sysboard/revn/ringname.chips
```

revn は **rev1** ディレクトリのコピーです。autoconfig は、必要に応じて、このコピーされたディレクトリで **chips** ファイルを編集します。

```
/common/board/sysboard/revn/template.chips
```

revn は **rev1** ディレクトリのコピーです。可変長リングでは、正しいチップ数が記述されているテンプレートが必要です。このテンプレートは **ringname.chips** ファイルにコピーすることができます。

```
/common/board/centerplane/revn/ringname.chips
```

revn は **rev1** ディレクトリのコピーです。autoconfig は、関係する各リングの **ringname.chips** ファイルを編集します。

名前	board_id - ボード ID の読み取りまたは書き込み
形式	<p>board_id [-r] -b <i>board_type</i> -n <i>board_number</i></p> <p>board_id -w -b <i>board_type</i> -n <i>board_number</i> -p <i>part_number</i> -s <i>serial_number</i></p>
注意	このコマンドはサン・マイクロシステムズの製造関係者のみが使用しません。
機能説明	<p>サン・マイクロシステムズの製造技術者は、board_id コマンドを使用することによって、システムボード、センタープレーン、センタープレーンのサポートボード、制御ボード、メモリーモジュール、または入出力モジュールに組み込まれた EEPROM 内のボード ID の表示 (読み取り) または割り当て (書き込み) を行うことができます。</p> <p>-r と -w のいずれも指定されていない場合は、-r が設定されます。</p>
オプション	<p>以下のオプションがサポートされています。</p> <p>-w (書き込み) 指定したパーツ番号とシリアル番号を、指定したボードに割り当てます。このオプションはサン・マイクロシステムズの製造関係者のみが使用します。</p> <p>-r (読み取り) 指定したボードについての情報を表示します。</p> <p>-b <i>board_type</i> 読み取りまたは書き込みの対象となるボードタイプを指定します。<i>board_type</i> には、次のいずれかの値を設定します。</p> <p>sb - システムボード cp - センタープレーン cb - 制御ボード csb - センタープレーンサポートボード mem - メモリーモジュール io - 入出力モジュール</p>

- n board_number** 読み取りまたは書き込みの対象となるボード番号を指定します。*board_number* には、**-b** オプションの指定に対応する次のいずれかの値を設定します。
 システムボード番号: 0 ~ 15
 センタープレーンハーフ: 0 または 1
 制御ボード: 0 または 1
 センタープレーンサポートボード: 0 または 1
 メモリーモジュール: 0 ~ 15
 入出力モジュール: 0 ~ 15
- p part_number** 指定したパーツ番号をボードに割り当てます。*part_number* は *ccc-aaaa-ss-rr* の形式で指定します。
ccc: 3 桁のカテゴリコード。現在、このコードの値は 612 です。
aaaa: 4 桁の基本パーツ番号 (アセンブリ番号)。
ss: 2 桁のパーツ番号の接尾辞。
rr: 2 桁のバージョン番号。
- s serial_number** 指定したシリアル番号をボードに割り当てます。*serial_number* は 3 ~ 12 文字の印刷可能な文字列です (スペースは含みません)。

使用例

例 1 board_id コマンドの使用

システムボード上のボード 1 についての情報を表示します。

```
board_id -b sb -n 1
```

システムボード上のボード 1 に、パーツ番号 501-2341-12-01 とシリアル番号 37KR59 を割り当てます。

```
board_id -w -b sb -n 1 -s 37KR59 -p 501-2341-12-01
```

センタープレーン上のボード 1 についての情報を表示します。**-r** も **-w** も指定されていないため、*board_id* は **-r** (デフォルト値) を採用します。**-r** は **-s** と **-p** を無視します。

```
board_id -b cp -n 1 -s 37KR59 -p 501-2341-12-01
```

名前	bringup - ドメインの設定と起動
形式	bringup [-f] [-F] [-p <i>proc</i>] [-Q <i>boot_proc</i>] [-aghvCL] [[-A] {on off}] [-l <i>level</i>] [[-D] {on off}] [-X <i>blacklist_file_pathname</i>] [<i>boot_args</i>]
機能説明	<p>bringup は以下のステップを実行して、環境変数 <code>SUNW_HOSTNAME</code> に指定されているドメインを起動します。いずれかのステップが失敗すると、bringup はエラーメッセージを表示して異常終了します。</p> <ol style="list-style-type: none"> 1. power(1M) コマンドを実行して、ドメインの電源が入っているかどうかをチェックします。電源が入っていれば次のステップに進み、入っていない場合は、ドメインの電源を入れるように促すメッセージを表示して異常終了します。-f が指定されている場合は、power(1M) コマンドは実行されません。 2. check_host(1M) を実行して、ドメインがすでに立ち上がっているかどうかを判別します。立ち上がっていない場合は、ドメインを起動します。すでに立ち上がっている場合は、bringup はそのことを知らせるメッセージを表示し、続行するかどうかを確認します。n を入力すると、bringup は終了します。y を入力すると、ドメインがハングしたために bringup を実行したのかどうかを確認するメッセージが表示されます。このメッセージに対して行った応答は、問題追跡のみを目的として記録されます。いずれの場合も、bringup はドメインを起動します。 3. すでに立ち上がっているか、立ち上げが行われているドメインが他にないかどうかをチェックして、次のステップで hpost(1M) にオプションの -c (センタープレーンの設定) を指定するかどうかを決定します。別のドメインの立ち上げが行われていて、センタープレーンを設定している最中である場合は、bringup はそのドメインのセンタープレーンの設定が終了するまで待つてから、処理を続行します。すでに立ち上がっているか、立ち上げが行われているドメインが他にない場合は、センタープレーンの設定の開始を知らせるメッセージが表示され、処理を続行するかどうかを確認します (コマンド行で -f を指定した場合は、bringup はこのメッセージを表示せずに処理を続行します)。 <ul style="list-style-type: none"> ■ y を入力すると、bringup はセンタープレーンを設定し、処理を続行します。センタープレーンが設定済みである場合 (たとえば以前のドメインによって設定されている場合) は、再設定が行われず。再設定しても問題はありますが、bringup の実行時間は少し長くなります。 ■ n を入力し、センタープレーンがすでに設定されている場合は、bringup はセンタープレーンを再設定せずに処理を続行します。 ■ n を入力した際にセンタープレーンが未だ設定されていない場合は、hpost(1M) は多くの場合失敗します。

注 - 立ち上がっているドメインがなく、センタープレーンが設定済みかどうか確かでない場合は、**y** を入力するのが一番安全です。

この時点で **bringup** は、ドメインが Solaris 2.6 オペレーティング環境よりも前のバージョンの Solaris で作成されたかどうかを判別します。Solaris 2.6 以前のバージョンの Solaris で作成されている場合は、次のステップ 4 を省略してステップ 5 に進みます。その他の場合は、ステップ 4 を実行してからステップ 5 に進みます。

4. ドメインごとに **domain_unlink(1M)** を実行して、それらのドメインが IDN のメンバーでないことを確認します。

その間以下のメッセージが表示されます。このメッセージは、ドメインの各 IDN のサイズおよびそれらのドメインメンバーの状態によって異なりますが、数秒間表示されます。

Checking *domain* IDN configuration...

確認作業が終了すると、**done** と表示されます。

IDN 操作中にエラーが発生すると、**error** と表示されると共に **domain_unlink(1M)** コマンドにより収集された出力が表示されます。別の IDN 操作が進行中の場合は、**busy** と表示され、前の IDN 操作が完了するまで待機します。

複数の IDN メンバーの状態が不明なために **domain_unlink(1M)** コマンドが失敗した場合は、**bringup** を実行する前に、それらの不明なドメインのリンクを解除するために、ドメインに対して 1 つずつ手動で **domain_unlink(1M)** コマンドを実行する必要があります。

各 IDN からのドメインのリンク解除は、一時的であり、MIB は更新されない点に注意してください。ドメインを起動すると、適切な IDN ソフトウェアが提供されている場合には、**edd(1M)** つまり SSP イベント検出デーモンがドメインの可用性を検出し、自動的に **domain_link(1M)** を実行してドメインを該当する IDN に再接続します。

5. **-c** オプションを付けて **hpost(1M)** を実行し、センタープレーンとドメインを続けて設定します。あるいは、**-c** オプションを付けずに **hpost(1M)** が実行された場合は、ドメインだけが設定されます。
6. **obp_helper(1M)** と **netcon_server(1M)** を起動し、OBP とオペレーティングシステムの起動処理を実行して、ネットワークコンソールを設定します。
7. 最終的なドメイン設定に従って MIB を更新します。

注 - **bringup** は、最初のドメインを起動する前に、最初のドメインのシステムクロックとクロックソースも設定します。

オプション

以下のオプションがサポートされています。

- f そのドメインがすでに立ち上がっている場合でも、bringup を強制的に実行します。
注意： コマンド行に指定されているかどうかに関係なく、bringup は必要に応じて -C を hpost(1M) に渡します。ただし、-C と一緒に -f を指定すると、bringup は他に立ち上がっているドメインのあるなしにかかわらず -C を渡します。この結果、すでに立ち上がっていたドメインがあれば、それらはすべてリセットされます。したがって、-C と -f を一緒に指定するときには、細心の注意が必要となります。-f を使用すると、bringup はドメインの電源が入っているかどうかをチェックしないので注意してください。
 - F このオプションはコマンド行では使用しないでください。このオプションは自動再起動スクリプト用に予約されています。edd(1M) を参照。
 - h bringup コマンド行オプションの簡単な説明を出力します。
 - Q より高速な、限定されたバージョンの POST を実行するように hpost(1M) に要求します。
-
- 注** - コマンド行で -Q を使用しないでください。-Q は、uadmin(2) システムコールまたは OpenBoot が開始したリセットの結果として、ドメインの再起動が要求されたとき、SSP ソフトウェア (edd(1M) 参照) によって起動されます。uadmin(2) システムコールは、reboot(1M)、uadmin(1M)、shutdown(1M)、および init(1M) コマンドによって使用されます。ドメインの状態によっては、手動で bringup -Q を実行すると、bringup で障害が発生することがあります。
-
- L -s と -v20 のオプションを使用するように hpost(1M) に要求します。この結果、出力はすべて syslog に送られます。
 - v -v70 オプションを使用するように hpost(1M) に要求します。この結果、より詳細な情報が作成されます。
 - S bringup は、このオプションを netcon_server(1M) に渡します。

boot_args 上記以外の引数はブート引数とみなされ、*obp_helper(1M)* に渡されます。*obp_helper(1M)* はこれらの引数を 1 つずつ **OpenBoot boot(1M)** コマンドに渡します。*boot_args* に他の引数を付ける場合は、必ず、引用符で囲んでください。たとえば、以下のようにします。

```
bringup "kadb -d"
"- -" の後の引数は、常に obp_helper(1M) に渡されます。これは bringup(1M) 引数と競合する引数、または " - " がついている名前を指定するに便利です。たとえば、以下の 2 つのコマンドは同等で、-D 引数は obp_helper(1M) に渡されます。
bringup -D
bringup - - -D
```

-a, -g, -llevel, -p proc, -C, -X blacklist_file_pathname
bringup はこれらのオプションを *hpost(1M)* に渡します。

-A {on | off}, -D {on | off}
bringup(1M) はこれらのオプションを *obp_helper(1M)* に渡します。

環境

環境変数 *SUNW_HOSTNAME* には、ドメイン名が設定されている必要があります。

関連項目

『Sun Enterprise 10000 SSP リファレンスマニュアル』の *check_host(1M)*, *domain_unlink(1M)*, *hpost(1M)*, *netcon_server(1M)*, *obp_helper(1M)*, *snmpd(1M)*

Solaris Reference Manual (英語) の *boot(1M)*

名前	cancelcmdsync, cmdsync, initcmdsync, savecmdsync - コマンド同期コマンド
形式	<p>cancelcmdsync <i>cmdsync_descriptor</i></p> <p>initcmdsync <i>script_name</i> [parameters]</p> <p>savecmdsync -M <i>identifier</i> <i>cmdsync_descriptor</i></p>
機能説明	<p>一連のコマンド同期コマンドは、連動して SSP フェイルオーバーによって割り込まれたユーザー定義のスキプトの回復を制御します。コマンドの同期を有効にするには、ユーザー定義のスキプトに以下のコマンドを挿入します。</p> <ul style="list-style-type: none"> ■ initcmdsync は、回復するスキプトを識別するためのコマンド同期記述子を生成します。 <p>この記述子は、コマンド同期リストに追加されます。この記述子によって、フェイルオーバーの後に新しいメイン SSP で再起動するスキプトとコマンドを識別します。</p> <ul style="list-style-type: none"> ■ savecmdsync は、フェイルオーバーの後に処理を再開できるスキプト上の場所を特定するマーカーを追加します。 ■ cancelcmdsync は、コマンド同期リストからコマンド同期記述子を削除します。削除することで、確実にスキプトは一度だけ実行され、次のフェイルオーバーの後には実行されません。 <p>スキプトのすべての終了パスに cancelcmdsync シーケンスを記述して、コマンド同期リストから記述子を確実に削除します。記述子を削除しない状態でフェイルオーバーが起動すると、スキプトは新しいメイン SSP で再実行されます。</p> <hr/> <p>注 - コマンドの同期を有効にするには、initcmdsync と cancelcmdsync の両方のシーケンスをスキプトに記述する必要があります。savecmdsync コマンドは省略可能ですが、処理を再開できるスキプト上の特定の場所を指定する場合のみ使用します。</p> <hr/>
オプション	<p>以下のオプションをサポートしています。</p> <p><i>cmdsync_descriptor</i> ユーザー定義のスキプトを識別するコマンド同期記述子を指定します。この記述子は、initcmdsync コマンドによって返される標準出力値です。</p> <p>-M <i>identifier</i> フェイルオーバーの後に再開できるスキプト上の場所を指定します。この識別子には、必ず正の整数を使用します。</p>

拡張機能説明

parameters ユーザー定義のスクリプトに関連するオプションとパラメタを指定します。これらのパラメタはスペア SSP に格納され、フェイルオーバーの後に指定されたスクリプトを再起動するときに使用されます。

script_name 同期させるユーザー定義のスクリプトの名前を指定します。

コマンド同期コマンドは、ユーザー定義スクリプトにおける決められた論理上の位置に挿入します。

たとえば Korn シェルスクリプトの場合は、以下のような構造になります。

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any interrupts.
# Use the cancelcmdsync to remove the script from the command
# synchronization list. Otherwise, the command will get restarted
# on the new main SSP.
#
clean_up () {
    cancelcmdsync $desc
    exit
}
# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point if provided
#
for arg in $*; do
    case $arg in
        -M )
            goto_label=$arg;;
        .
        .
    esac
done
# Place this script and all its parameters in the command synchronization
# list, which indicates the commands to be restarted after an SSP failover.
#
# NOTE: The script must be executable by user ssp and reside in the same
# directory on both the main and the spare SSP. If the command is not
# part of the defined PATH for user ssp, the absolute filename must be
# passed with the initcmdsync command.
#
initcmdsync script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been processed or an
# error is detected.
#
while (( $goto_label != 0 )) ; do
    #
    # Each case should represent a synchronization point in the script.
    #
    case $goto_label in
        #
        # Step 1: Do something
        #
        1 )
            do_something
            .
            .
            # Execute the savecmdsync command with the script's
            # descriptor and a unique marker to save the position.
            # If a failover occurs here, the commands
            # represented in the next goto_label (2) will be
```

```

        # resumed.
        #
        savecmdsync -M $(( $goto_label + 1 )) $desc
        goto_label=$(( $goto_label + 1 ))
        ;;
#
# Step 2: Do more things
#
2 )
    do_more_things
    .
    .
    savecmdsync -M $(( $goto_label + 1 )) $desc
    goto_label=$(( $goto_label + 1 ))
    ;;
#
# Step 3: Finish the last step and set the goto_label to 0
# so that the script ends.
3 )
    finish_last_step
    .
    .
    goto_label=0
    ;;
    esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsync $desc

```

終了ステータス

以下の終了ステータスが返されます。

0	正常に終了しました。
0 以外	エラーが発生しました。

注 - `initcmdsync` に対する標準出力は、コマンド同期記述子で構成されます。また、フェイルオーバーが起動しない場合（フェイルオーバーの後、またはシングル SSP 環境の場合）は、同期コマンドを含むスクリプトによってプラットフォームログファイルにエラーメッセージが生成され、0 以外の終了コードが返されます。このエラーメッセージは、無視しても問題はありません。

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

関連項目

runcmdsync(1M), showcmds(1M)

ディレクトリ /opt/SUNWssp/examples/cmdsync の (同期コマンドを含む) ユーザー定義スクリプトの例

名前	cb_prom - PROM のアップグレード
形式	<pre>cb_prom -d file [-f] -h hostname cb_prom -e -h hostname cb_prom -p file [-h hostname] cb_prom -r -h hostname</pre>
注意	このコマンドは、購入先の技術者のみが使用します。
機能説明	cb_prom は、主制御ボードに組み込まれている Sun Enterprise 10000 制御ボードのフラッシュ PROM の内容を操作します。この PROM では、制御ボードエグゼクティブ (CBE) をダウンロードするのに必要なブートファームウェアに現場でアップグレードすることができます。
オプション	<p>以下のオプションをサポートしています。</p> <pre>-d file [-f] フラッシュ PROM の内容を、指定したファイル (file) に ダンプします。デフォルトでは、ファイルがすでに存在 する場合は、-d によりファイルは上書きされません。-f、 つまり強制オプションを使用すると、この保護機能を無 効にできます。 -e フラッシュ PROM の内容を消去します。 -h hostname 指定したホスト名 (hostname) の制御ボード上で cb_prom コマンドを実行します。 -p file 指定したファイル (file) の内容を使ってフラッシュ PROM をプログラミングします。 -r 制御ボード上で現在使用中のフラッシュ PROM のバー ジョンを表示します。</pre>
注意事項	ブートファームウェアのアップグレードまたは消去が有効になるのは、cb_reset(1M) などによる制御ボードの再起動後のみです。

名前	cb_reset - ハングした制御ボードのリセットと再起動
形式	cb_reset [-v] [-p <i>platform_name</i>] [<i>control_board_name</i>]
機能説明	デフォルトの書式では、 cb_reset は、環境変数 SUNW_HOSTNAME で指定した Sun Enterprise 10000 システムの制御ボードに固定 Ethernet パケットを送ります。これによって、ボードが自動的にリセットされ再起動されます。 cb_reset は、立ちあがっているすべてのネットワークリンクにリセットパケットを送ります。 ifconfig(1M) に -au オプションを指定して使用すると、立ちあがっているリンクの一覧を表示できます。
オプション	以下のオプションをサポートしています。 <p>-v (Verbose モード) パケットが送信されたことを確認するメッセージを表示します。このメッセージには、発信元 (パケットを送った SSP) と着信先 (パケットを受け取った制御ボード) の Ethernet アドレスも表示されます。このオプションは、コマンド行の最初に指定する必要があります。</p> <p>-p <i>platform_name</i> 指定した Sun Enterprise 10000 システム (<i>platform_name</i>) の制御ボードに Ethernet パケットを送ります。</p> <p><i>control_board_name</i> 指定した制御ボード (<i>control_board_name</i>) に Ethernet パケットを送ります。<i>control_board_name</i> の名前は cb_config(4) で指定します。一緒に -p を指定した場合は、cb_reset は指定された Sun Enterprise 10000 システムの指定された制御ボードにパケットを送ります。-p が指定されていない場合は、cb_reset は環境変数 SUNW_HOSTNAME で指定されたシステムの指定制御ボードにパケットを送ります。</p>
ファイル	以下のファイルがサポートされます。 <p>\$SSPVAR/.ssp_private/cb_config 制御ボードの設定ファイル</p>
関連項目	edd(1M) , ssp_startup(1M)

名前	cbs - 制御ボードサーバー								
形式	cbs								
注意	このコマンドは手動では実行しないでください。このコマンドは SSP スタートアップスクリプトによって自動的に呼び出され、必要に応じて再起動されます。								
機能説明	<p>cbs サーバープログラムは、SSP で実行されているクライアントプログラムが、Sun Enterprise 10000 の制御ボードにアクセスできるようにします。cbs はさらに、Sun Enterprise 10000 の JTAG 走査データベースへ単独でアクセスするほか、すべての JTAG 操作のロックマネージャーともなります。</p> <p>cbs はクライアントの要求を CBMP (Control Board Management Protocol) メッセージに変換し、Sun Enterprise 10000 の制御ボードで動作している制御ボードエグゼクティブ (CBE) に渡します。cbs は domain_config(4) と cb_config(4) に基づき、管理すべきプラットフォームを判別し、通信に使用する制御ボードを判別します。</p> <p>CBE と cbs の間の通信プロトコルによって、SSP アプリケーションは、制御ボードのリソースの検索と変更、JATG 走査、監視スクリプトの実行要求を行います。</p> <p>各 SSP が一度に実行できる cbs のインスタンスは 1 つだけです。</p>								
ファイル	<p>以下のファイルがサポートされます。</p> <table border="0"> <tr> <td>\$SSPVAR/data/Ultra-Enterprise-10000</td> <td>Sun Enterprise 10000 の JTAG 走査データベース</td> </tr> <tr> <td>\$SSPVAR/pids/cbs.pid</td> <td>プロセス ID ファイル</td> </tr> <tr> <td>\$SSPVAR/.ssp_private/domain_config</td> <td>ドメイン設定ファイル</td> </tr> <tr> <td>\$SSPVAR/.ssp_private/cb_config</td> <td>制御ボード設定ファイル</td> </tr> </table>	\$SSPVAR/data/Ultra-Enterprise-10000	Sun Enterprise 10000 の JTAG 走査データベース	\$SSPVAR/pids/cbs.pid	プロセス ID ファイル	\$SSPVAR/.ssp_private/domain_config	ドメイン設定ファイル	\$SSPVAR/.ssp_private/cb_config	制御ボード設定ファイル
\$SSPVAR/data/Ultra-Enterprise-10000	Sun Enterprise 10000 の JTAG 走査データベース								
\$SSPVAR/pids/cbs.pid	プロセス ID ファイル								
\$SSPVAR/.ssp_private/domain_config	ドメイン設定ファイル								
\$SSPVAR/.ssp_private/cb_config	制御ボード設定ファイル								
関連項目	machine_server(1M), domain_config(4), cb_config(4)								

名前	check_host - ドメインが立ち上がっているかどうかの判別
形式	check_host [-q]
機能説明	<p>check_host は bringup(1M) スクリプトから呼び出され、環境変数 SUNW_HOSTNAME に指定されているドメインが立ち上がっているかどうかをチェックします。設定済みのすべてのプロセッサがオペレーティングシステムを実行している場合に、check_host はそのドメインが立ち上がっているものとみなし、「Host is UP」というメッセージを表示したうえで、ステータス 0 を返します。設定済みのいずれかのプロセッサがオペレーティングシステムを実行していない場合は、check_host はそのドメインがダウンしているものとみなし、「Host is DOWN」というメッセージを表示したうえで、ステータス 1 を返します。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>-q (Quiet) 終了コードを報告しますが、標準出力には情報をエコーしません。</p>
注意事項	<p>プロセッサは hpost(1M) によって設定されます。hpost(1M) は bringup(1M) によって実行されます。</p> <p>ドメインがすでに立ち上がっている場合は、bringup(1M) コマンドは実行されません。この安全上の機能を無効にしたい場合は、bringup -f を使います。</p>
関連項目	bringup(1M), edd(1M), hpost(1M), snmpd(1M)

名前	cmdsync, cancelcmdsync, initcmdsync, savecmdsync - コマンド同期コマンド
形式	cancelcmdsync <i>cmdsync_descriptor</i> initcmdsync <i>script_name</i> [parameters] savecmdsync -M <i>identifier cmdsync_descriptor</i>
機能説明	<p>一連のコマンド同期コマンドは、連動して SSP フェイルオーバーによって割り込まれたユーザー定義のスキ립トの回復を制御します。コマンドの同期を有効にするには、ユーザー定義のスキ립トに以下のコマンドを挿入します。</p> <ul style="list-style-type: none"> ■ initcmdsync は、回復するスキ립トを識別するためのコマンド同期記述子を生成します。 <p>この記述子は、コマンド同期リストに追加されます。この記述子によって、フェイルオーバーの後に新しいメイン SSP で再起動するスキ립トとコマンドを識別します。</p> <ul style="list-style-type: none"> ■ savecmdsync は、フェイルオーバーの後に処理を再開できるスキ립ト上の場所を特定するマーカーを追加します。 ■ cancelcmdsync は、コマンド同期リストからコマンド同期記述子を削除します。削除することで、確実にスキ립トは一度だけ実行され、次のフェイルオーバーの後には実行されません。 <p>スキ립トのすべての終了パスに cancelcmdsync シーケンスを記述して、コマンド同期リストから記述子を確実に削除します。記述子を削除しない状態でフェイルオーバーが起動すると、スキ립トは新しいメイン SSP で再実行されます。</p>
	<p>注 - コマンドの同期を有効にするには、initcmdsync と cancelcmdsync の両方のシーケンスをスキ립トに記述する必要があります。savecmdsync コマンドは省略可能ですが、処理を再開できるスキ립ト上の特定の場所を指定する場合のみ使用します。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>cmdsync_descriptor ユーザー定義のスキ립トを識別するコマンド同期記述子を指定します。この記述子は、initcmdsync コマンドによって返される標準出力値です。</p> <p>-M identifier フェイルオーバーの後に再開できるスキ립ト上の場所を指定します。この識別子には、必ず正の整数を使用します。</p>

拡張機能説明

parameters ユーザー定義のスクリプトに関連するオプションとパラメータを指定します。これらのパラメータはスペア SSP に格納され、フェイルオーバーの後に指定されたスクリプトを再起動するときに使用されます。

script_name 同期させるユーザー定義のスクリプトの名前を指定します。

コマンド同期コマンドは、ユーザー定義スクリプトにおける決められた論理上の位置に挿入します。

たとえば Korn シェルスクリプトの場合は、以下のような構造になります。

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any interrupts.
# Use the cancelcmdsnc to remove the script from the command
# synchronization list. Otherwise, the command will get restarted
# on the new main SSP.
#
clean_up () {
    cancelcmdsnc $desc
    exit
}
# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point if provided
#
for arg in $*; do
    case $arg in
        -M )
            goto_label=$arg;;
        .
        .
    esac
done
# Place this script and all its parameters in the command synchronization
# list, which indicates the commands to be restarted after an SSP failover.
#
# NOTE: The script must be executable by user ssp and reside in the same
# directory on both the main and the spare SSP. If the command is not
# part of the defined PATH for user ssp, the absolute filename must be
# passed with the initcmdsnc command.
#
initcmdsnc script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been processed or an
# error is detected.
#
while (( $goto_label != 0 )) ; do
    #
    # Each case should represent a synchronization point in the script.
    #
    case $goto_label in
        #
        # Step 1: Do something
        #
        1 )
            do_something
            .
            .
            # Execute the savecmdsnc command with the script's
            # descriptor and a unique marker to save the position.
            # If a failover occurs here, the commands
            # represented in the next goto_label (2) will be
```

```

        # resumed.
        #
        savecmdsync -M $(( $goto_label + 1 )) $desc
        goto_label=$(( $goto_label + 1 ))
        ;;
#
# Step 2: Do more things
#
2 )
    do_more_things
    .
    .
    savecmdsync -M $(( $goto_label + 1 )) $desc
    goto_label=$(( $goto_label + 1 ))
    ;;
#
# Step 3: Finish the last step and set the goto_label to 0
# so that the script ends.
3 )
    finish_last_step
    .
    .
    goto_label=0
    ;;
    esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsync to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsync $desc

```

終了ステータス

以下の終了ステータスが返されます。

0	正常に終了しました。
0 以外	エラーが発生しました。

注 - `initcmdsync` に対する標準出力は、コマンド同期記述子で構成されます。また、フェイルオーバーが起動しない場合 (フェイルオーバーの後、またはシングル SSP 環境の場合) は、同期コマンドを含むスクリプトによってプラットフォームログファイルにエラーメッセージが生成され、0 以外の終了コードが返されます。このエラーメッセージは、無視しても問題はありません。

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

関連項目

runcmdsync(1M), showcmdsync(1M)

ディレクトリ /opt/SUNWssp/examples/cmdsync の (同期コマンドを含む) ユーザー定義スクリプトの例

名前	datasyncd - データ同期デーモン
形式	datasyncd
注意事項	このデーモンは手動で実行しないでください。このデーモンは SSP 起動スクリプトによって呼び出され、必要に応じて監視・再起動されます。
機能説明	<p>データ同期デーモンは、メイン SSP とスペア SSP 間で SSP 設定ファイルの同期をとります。構成ファイルになんらかの変更が行われると、データ同期デーモンは私設 SSP データネットワーク上で TCP/IP 接続を行って、メイン SSP からスペア SSP へ構成ファイルをコピーします。datasyncd からのトラフィックは、制御ボードの管理に使用されていない専用接続によって経路指定されます。</p> <p>datasyncd は、SSP 構成ファイル (\$SSPVAR/.ssp_private/fad_files) にリストされたプラットフォームとドメインに関するすべてのファイルだけでなく、データ伝播リスト (\$SSPVAR/.ssp_private/user_file_list) で指定されたユーザー定義ファイルも同期させます。このリストで、メイン SSP からスペア SSP へコピーするユーザーファイルを判断します。ユーザー定義ファイルは、ユーザー ssp による読み取りと書き込みが可能である必要があります。このデータ伝播リストは、setdatasync(1M) コマンドで操作します。</p> <p>メイン SSP 上の SSP 構成ファイルが変更されると、datasyncd は変更されたファイルを直ちにスペア SSP へコピーします。また、datasyncd はメイン SSP 上のユーザー定義ファイルに対する変更を 1 時間ごとに (またはユーザーが setdatasync コマンドで指定した時間間隔で) 監視します。最後に伝播された後でユーザーファイルが変更されると、datasyncd がスペア SSP にファイルをコピーします。</p> <p>datasyncd は、SSP 構成ファイルまたはユーザーファイルがスペア SSP 上にあるかどうかをチェックしません。また、ユーザーがスペア SSP 上の SSP 構成ファイルまたはユーザーファイルに対して行う変更についても監視しません。したがって、ユーザーがスペア SSP からユーザー定義ファイルまたは SSP 構成ファイルを削除しても、メイン SSP 上に保存されている同じファイルが変更されない限り、そのファイルが自動的にスペア SSP に再伝播されることはありません。</p> <p>datasyncd は fod(1M) や fad(1M) などの SSP デーモンに従います。datasyncd デーモンはメイン SSP でのみ実行されます。</p>

ファイル

以下のファイルがサポートされます。

<code>\$SSPVAR/.ssp_private/fad_files</code>	設定ファイルリスト
<code>\$SSPVAR/.ssp_private/user_file_list</code>	データ伝播リスト
<code>/tmp/ds_backup.cpio</code>	データ同期のバックアップファイル

関連項目

`fad(1M)`、`fod(1M)`、`setdatasync(1M)`、`showdatasync(1M)`、`fad_files(4)`

名前	domain_create - 新規のドメインの作成または既存のドメインの再作成
形式	domain_create -d <i>domain_name</i>
機能説明	<p>domain_create -d <i>domain_name</i> [-b <i>sysboard_list</i>] [-o <i>os_version</i>] [-p <i>platform_name</i>] [-t <i>platform_type</i>]</p>
機能説明	<p>domain_create コマンドを使用して、ドメインを作成したり、domain_remove(1M) によって削除されたドメインを再作成することができます。domain_remove(1M) によって削除されたドメインのリストを表示するには、domain_history(1M) コマンドを使用します。</p> <p>新しいドメインを作成する場合は、必ず -d、-b、-o、-p オプションを指定します。domain_create は新しいドメインを作成し、-d で指定された名前を付けます。</p> <p>ドメインを元の構成のまま再作成する場合は、-d オプションのみを指定してください。domain_create は元の指定に従ってドメインを再作成します。削除前のドメインに関する情報を取得できない場合は、domain_create はエラーを返します。</p>
注	<hr/> <p>注 - コマンド行にオプションのパラメタを指定しない場合は、コマンドは domain_history ファイルからドメインの元の指定を取得します。domain_history ファイルに元の指定に関する情報が見つからない場合は、エラーを返します。</p> <hr/>
注	<p>domain_create は、コマンド行から実行するか、または Hostview の中で Configuration -> Domain -> Create を選択して実行します (hostview(1M) を参照)。</p>
注	<p>削除する前のドメインの IDN メンバーシップに関する履歴情報は保存されていません。ドメインを作成する場合は、そのドメインは IDN に所属しません。</p>
条件	<p>domain_create を実行できるのは、以下の条件がすべて満たされた場合です。</p> <ul style="list-style-type: none"> ■ ドメイン用に指定しようとするシステムボードに、少なくともネットワークインタフェースと SCSI インタフェースが備わっており、独立したシステムをサポートできるだけの十分なメモリーが備わっている。 ■ ホストの /etc/hosts ファイルにドメイン名が定義されている。 ■ 指定したシステムボードがすべて存在しており、かつ未使用である。

オプション

以下のオプションをサポートしています。

- d *domain_name* *domain_name* に指定された削除済みドメインを再作成します。または新しいドメインを作成し、*domain_name* に指定された名前を付けます。
- b *sysboard_list* 指定したシステムボードを新しいドメインに組み込みます。システムボードはボード番号によって示し、複数のボードはスペースで区切ります。
- o *os_version* ドメインで実行される SunOS のオペレーティングシステムのバージョン (例 5.5.1、5.6、5.7、5.8) を指定します。
- p *platform_name* 新しいドメインとして使用するボードが装着されているプラットフォーム (Sun Enterprise 10000 システム) の名前を指定します。
- t *platform_type* ドメインが元々属しているシステムの種類を指定します。デフォルトは、ドメインが元々属しているマシンのプラットフォームタイプです。Sun Enterprise 10000 システムのプラットフォームタイプは Ultra-Enterprise-10000 です。

使用例

例 1 ドメインの作成

以下のコマンドを使用すると、**junior** という名前のドメインが作成されます。このドメインは、番号 0、2、9 のボードから構成され、バージョン 5.5.1 のオペレーティングシステムを実行します。**junior** のボードは、**dad** というプラットフォーム名のマシンのものです。

```
domain_create -d junior -b 0,2,9 -o 5.5.1 -p dad
```

戻り値

`domain_create` が正常に完了した場合は、値 0 が返され、SSP が再構成されて新しいドメインが表示されます。正常に完了しなかった場合は、0 以外の値が返されます。

注意事項

`domain_create` コマンドが一度に作成できるドメインは 1 つだけです。

コマンド行からドメインを作成 (Hostview を使用してドメインを作成した場合、この作業は必要ありません) したら、SSP コマンドが正常に動作するように、環境変数 `SUNW_HOSTNAME` を次のように設定してください。`domain_switch(IM)` コマンドを使用すれば、環境変数を設定できます。

環境変数を設定したら、bringup(1M) コマンドを使用してこのドメインを立ち上げることができます。

オペレーティングシステムの新しいパッチには、ホスト依存の SSP バイナリの新しいバージョンが含まれることがあります。新しいパッチを実行するドメインを作成する際に使用するオペレーティングシステムのバージョン番号については、そのパッチの README ファイルを参照してください。

関連項目

domain_history(1M), domain_remove(1M), domain_rename(1M), domain_status(1M), domain_switch(1M)

名前	domain_history - ドメインの履歴の表示または削除
形式	domain_history [-d <i>domain_name</i>] [-r]
機能説明	オプションを指定せずに domain_history を実行すると、 domain_history(4) ファイルが表示されます。このファイルには、 domain_remove(1M) によって削除されたすべてのドメインの設定情報が含まれています。
オプション	以下のオプションをサポートしています。 -d <i>domain_name</i> 指定したドメインの設定情報だけを表示します。 -r domain_history(4) ファイルからすべての設定情報を削除します。-d を一緒に指定した場合は、指定したドメインの設定情報だけが削除されます。 <hr/> 注 - 削除したドメインの再作成は新しいドメインの作成よりも容易です。したがって、-r オプションの使用に過度に慎重になる必要はありません。 <hr/>
戻り値	domain_history が正常に完了した場合は、値 0 が返されます。正常に完了しなかった場合は、0 以外の値が返されます。
使用例	例 1 設定情報の表示 以下のコマンドを実行すると、以前存在していた junior という名前のドメインに関する設定情報が表示されます。 <pre>domain_history -d junior</pre> 例 2 履歴ファイルのエントリの削除 以前存在していた junior という名前のドメインに対するエントリを、 domain_history(4) ファイルから削除します。 <pre>domain_history -d junior -r</pre>
エラー	<i>domain_name</i> does not exist. 指定された名前が domain_history(4) ファイルの中に見つかりません。

domain_history(1M)

保守コマンド

関連項目

domain_create(1M), domain_remove(1M), domain_rename(1M),
domain_status(1M), domain_switch(1M), domain_history(4)

名前	domain_link - ドメインのリンクによる InterDomain Networks (IDN) の形成・拡張
形式	domain_link <i>domain_name_a</i> <i>domain_name_b</i>
機能説明	<p>domain_link コマンドを使用すると、Sun Enterprise 10000 サーバー上だけで動的システムドメインをリンクすることができます。この動的システムドメインを本書ではドメインと呼びます。domain_link では、同じサーバー上で IDN を作成・拡張することができます。ドメインのメモリのホストとなっているシステムボードには、少なくとも 1 つのプロセッサが存在する必要があります。指定されたドメインの 1 つがすでに IDN のメンバーになっている場合は、他の指定されたドメインがその IDN に追加されます。両方のドメインがそれぞれ別の IDN のメンバーである場合には、domain_link コマンドによりそれらの IDN は組み合わせられて 1 つのより大きな IDN になります。どちらのドメインも IDN に属さない場合は、両方のドメインがそのメンバーとなる新しい IDN が作成されます。</p> <p>domain_link(1M) コマンドで指定できるドメイン名は、2 つだけです。マルチドメイン IDN を作成する場合は、まず 2 つのドメインをリンクし、それからその IDN に別のドメインをリンクします。別のドメインをリンクするには、domain_link(1M) コマンドに IDN 内のいずれかのドメイン名と新しいドメイン名を一緒に指定します。</p> <p>各 IDN には、マスタートドメインが 1 つあります。マスタートドメインは、『Sun Enterprise 10000 InterDomain Networks ユーザーマニュアル』で説明されている共有メモリ領域 (SMR) を管理します。マスタートドメインは、IDN サブシステムにより動的に選択されます。</p> <hr/> <p>注 - ドメインがすでに立ち上がって動作している場合は、ドメインを IDN にリンクする前に、OBP プロンプトで idn-smr-size 変数を設定し、ドメインを再起動する必要があります。</p> <hr/> <p>ドメインの 1 つがダウンした場合でも、他のドメインは IDN を介して通信を続行できます。IDN 内のすべてのドメインは論理的にポイントツーポイント構成で接続されていますが、IDN 機能はネットワークインタフェースレベルでのブロードキャスト通信もサポートします。アービトレーション停止 (arbstop) および IDN の詳細は、『Sun Enterprise 10000 InterDomain Networks ユーザーマニュアル』を参照してください。</p>

オプション

以下のオプションをサポートしています。

domain_name_a リンクする最初のドメイン名を表します。ドメインには、既存の IDN のメンバー、またはまだ IDN にリンクされていないドメインを指定することができます。

domain_name_b リンクする 2 番目のドメイン名を表します。ドメインには、既存の IDN のメンバー、またはまだ IDN にリンクされていないドメインを指定することができます。

終了ステータス

以下の終了ステータスが返されます。

- 0 ドメインのリンクに成功。
- 0 以外 ドメインのリンクに失敗。

環境

IDN を作成する前、またはドメインを既存の IDN にリンクする前に、SUNW_HOSTNAME シェル環境変数が、IDN 機能をサポートするドメイン名に設定されていることを確認してください。この変数に関しては、プラットフォーム名では IDN コマンドは動作しません。

関連項目

domain_create(1M), domain_unlink(1M), domain_history(1M), domain_remove(1M), domain_rename(1M), domain_status(1M)

『Sun Enterprise 10000 InterDomain Networks ユーザーマニュアル』

名前	domain_remove - 既存のドメインの削除
形式	domain_remove [-d <i>domain_name</i>]
機能説明	<p>domain_remove は、指定されたドメインを削除し、そのシステムボードを Sun Enterprise 10000 上の「存在し、かつ未使用のシステムボード」として再設定します。ドメインを削除するには、そのドメインをシャットダウンする必要があります。また IDN のメンバーであるドメインを削除することはできません。以下の「エラー」セクションの「注意」を参照してください。</p> <p>domain_remove は以下のメッセージを表示して、特定の情報を保存するためのオプションを提供します。</p> <pre>The following subdirectories contain domain-specific information, such as messages files, configuration files, and hpost dump files. You may choose to keep these directories if you still need this information. This domain may be recreated with or without this information being saved.</pre> <p>domain_remove は特定のサブディレクトリを保存するかどうかを尋ね、ユーザーの選択に応じて処理を続行します。</p> <p>domain_remove が完了すると、解放されたボードを domain_create(1M) や DR Attach で使用できるようになります。</p> <p>domain_remove はコマンド行から実行できる以外に、Hostview でも実行できます。Hostview を使ってドメインを削除するには、そのドメインのいずれかのボードを選択し、Configuration -> Domain -> Remove の順で選択します。オペレーティングシステムを実行中のドメインは削除できません。</p> <p>設定検査の段階で操作を中断すると、domain_remove は異常終了します。この段階を過ぎて、再設定が開始されると、コマンドは中断できなくなります。</p>
戻り値	domain_remove が正常に完了した場合は、値 0 が返されます。正常に完了しなかった場合は、0 以外の値が返されます。
エラー	<p>指定したドメインがオペレーティングシステムを実行中です。</p> <pre>Domain '<i>domain_name</i>' is up, must be shutdown before removing. Continue anyway (y/n)?</pre> <p>注意 -domain_remove の処理を続行すると、オペレーティングシステムを実行中のドメインが削除されます。その場合は、ドメインがクラッシュし、データが失われる可能性があります。</p>

指定したドメインが IDN のメンバーです。ドメインを削除する前に IDN へのリンクを解除する必要があります。

```
domain_remove: Domain (domain_name)
is linked to an IDN. Domain must be unlinked before
it can be removed. See domain_unlink(1M).
```

注意事項

domain_remove コマンドが一度に削除できるドメインは 1 つだけです。

設定が削除されると、その設定に対応する obp_helper(1M) と netcon_server(1M) デーモンも終了します。

関連項目

domain_create(1M), domain_history(1M), domain_unlink(1M), domain_rename(1M), domain_status(1M), domain_switch(1M), netcon_server(1M)

名前	domain_rename - 既存のドメインの名前変更
形式	domain_rename -d <i>domain_name</i> [-n <i>new_domain_name</i>] [-o <i>new_os_version</i>]
機能説明	<p>domain_rename は、ドメインの名前とバージョンの一方、あるいは両方を変更します。-n オプションを指定すると、domain_rename は、-d オプションで指定されたドメインの名前を -n オプションで指定された名前に変更します。-o オプションを指定すると、domain_rename は、指定されたドメインの SunOS オペレーティングシステムのバージョン番号を変更します。</p> <p>domain_rename はコマンド行から実行できる以外に、Hostview でも実行できます。ターゲットのドメインがオペレーティングシステムを実行しているときは、コマンドは失敗します。</p> <p>ドメイン名を変更した場合は、インターネットアドレスや Sun Enterprise 10000 システムドメインおよび SSP 上のその他の設定情報を変更し、ドメインの新しい名前をシステムに認識させる必要があります。これを行わないと、SSP がそのドメインを制御できなくなります。</p> <p>構成検査の段階で操作を中断すると、domain_rename は異常終了します。この段階を過ぎて、再構成が開始されると、コマンドは中断できなくなります。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>-d <i>domain_name</i> 変更するドメインの名前</p> <p>-n <i>domain_name</i> -d <i>domain_name</i> で指定されたドメインの新しい名前</p> <p>-o <i>new_os_version</i> 指定されたドメインで実行されている SunOS オペレーティングシステムのバージョン (例 5.5.1、5.6、5.7、5.8)</p>
戻り値	domain_rename が正常に完了した場合は、値 0 が返されます。正常に完了しなかった場合は、0 以外の値が返されます。正常に完了した場合は、そのドメインを bringup(1M) コマンドで立ち上げることができます。
エラー	<p><i>domain_name</i> does not exist.</p> <p>指定された名前が見つかりません。</p>

new_domain_name already exists.

すでに使用されているドメイン名を割り当てようとした。

注意事項

コマンド行からドメイン名を変更した場合は、SSP コマンドの動作が適切になるように、環境変数 `SUNW_HOSTNAME` を次のように設定してください。

```
setenv SUNW_HOSTNAME new_domain_name
```

関連項目

`domain_create(1M)`, `domain_history(1M)`, `domain_remove(1M)`,
`domain_status(1M)`, `domain_switch(1M)`

名前	domain_status - ドメインの状態の表示
形式	domain_status [-d <i>domain_name</i>] [-m]
機能説明	domain_status は domain_config(4) ファイルを表示します。このファイルには、構成済みの各ドメインについて、ドメイン名、プラットフォームタイプ、プラットフォーム名、オペレーティングシステム、システムボード、および InterDomain Networks (IDN) についての情報が格納されています。
オプション	<p>以下のオプションをサポートしています。</p> <p>-d <i>domain_name</i> 指定したドメインの情報だけを表示します。</p> <p>-m ドメイン上で実行されている動的再構成 (DR) モデル (2.0、3.0、unknown のいずれか) を表示します。</p> <ul style="list-style-type: none"> ■ DR モデル 2.0 は、ドメインで行われている DR 操作が dr(1M) デーモンによって制御されていることを示します。 ■ DR モデル 3.0 は、ドメイン構成サーバー dcs(1M) がドメインの DR 操作を制御していることを示します。DR model 3.0 ドメインでは、Reconfiguration Coordination Manager (RCM) を使用して、ドメイン上で実行されているデータベース、クラスタ、ボリューム管理ソフトウェアなどのアプリケーションに対して DR 操作をコーディネートすることもできます。 ■ Unknown は、このドメインが立ち上がっていない、あるいはこのドメインに対して DR が正しく構成されていないため、DR モデル が調べられないことを示します。
戻り値	domain_status が正常に完了した場合は、値 0 が返されます。正常に完了しなかった場合は、0 以外の値が返されます。
注意事項	<p>すべての InterDomain Networks は、状態リスト中の番号によって識別され、番号の後にそのネットワークに属するドメインの名前が表示されます。以下に示すリスト番号はリストのためのタグにすぎず、IDN 固有の識別番号ではありません。</p> <p>0: xf6-b6 xf6-b7 xf6-b8 1: xf6-b3 xf6-b4</p>

表示されるドメインの状態情報には、ドメイン名、タイプ、プラットフォーム名、DR モデル のバージョン (domain_status コマンドに -m オプションを指定した場合のみ表示)、オペレーティングシステムのバージョン、システムボードがあります。

カラム内に表示されたエントリがそのカラムに割り当てられている幅よりも長い場合、このエントリと、隣のカラムのエントリとの間はスペースで区切られますが、それ以降のカラムに表示される情報はそれぞれのカラム見出しと位置が揃わなくなります。

使用例

例 1 ドメイン状態の表示例

```
% domain status -m
DOMAIN          TYPE                PLATFORM DR-MODEL  OS    SYSBDS
xf6-b2          Ultra-Enterprise-10000 all-xf6  2.0      5.6    2
xf6-b1          Ultra-Enterprise-10000 all-xf6  3.0      5.8    1
xf6-b5          Ultra-Enterprise-10000 all-xf6  2.0      5.6    5 13
xf6-b6          Ultra-Enterprise-10000 all-xf6  2.0      5.7    6 14
xf6-b7          Ultra-Enterprise-10000 all-xf6  3.0      5.8    7 15
xf6-b8          Ultra-Enterprise-10000 all-xf6  2.0      5.7    0 8
xf6-b4          Ultra-Enterprise-10000 all-xf6  3.0      5.8    12
xf6-b3          Ultra-Enterprise-10000 all-xf6  2.0      5.7    4
```

関連項目

domain_create(1M), domain_history(1M), domain_remove(1M), domain_rename(1M), domain_switch(1M)

名前	domain_switch - SUNW_HOSTNAME に指定されているドメイン名の変更
形式	domain_switch <i>domain_name</i>
機能説明	<p>domain_switch は、ユーザー ssp の .cshrc ファイルに定義されている csh(1) のエイリアスです。ユーザー ssp でログインした場合は、domain_switch をコマンド行から実行して、環境変数 SUNW_HOSTNAME を変更し、SSP セッションの制御対象となるドメイン名を指定できます。SUNW_HOSTNAME の新しい値を反映して、csh プロンプトも変更されます。</p> <p>domain_switch エイリアスは実際の SSP コマンドほど堅牢ではありません。複数の <i>domain_name</i> 引数を指定した場合には、最初の引数だけが使用されます。指定されたドメイン名が有効かどうかの検査は行われません。domain_name 引数を指定しないと、domain_switch は次のようなメッセージを表示します。</p> <pre>xf1-ssp:xf1% domain_switch Bad ! arg selector</pre>
使用例	<p>例 1 ドメイン名の変更</p> <div style="border: 1px solid black; padding: 5px;"><pre>xf1-ssp:xf1% domain_switch xf1-deux Switch to domain xf1-deux xf1-ssp:xf1-deux%</pre></div>

名前	domain_unlink - InterDomain Networks (IDN) からのドメインのリンク解除
形式	domain_unlink [-f -F] <i>domain_name</i> ...
機能説明	<p>domain_unlink コマンドは、IDN から 1 つまたは複数のドメインのリンクを解除します (このコマンドの 1 つのインスタンスでは、異なる IDN にリンクされた複数のドメインを削除することはできません)。IDN に残ったドメインは、ネットワークを介して通常の方法で通信を続行できます。ドメインのリンクを解除する前に、関連するネットワークインタフェースが使用中でないことを確認してください。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>-f ソフトな強制オプションを表します。domain_unlink でこのオプションを指定すると、指定されたドメインすべてのリンクが通常の方法で解除されますが、ドメインからまったく応答がないためにタイムアウトした場合には、domain_unlink は -F オプションを使用してそのリンクを削除し、ドメインのリンクを強制的に解除します。</p> <p>-F ハードな強制オプションを表します。domain_unlink でこのオプションを指定すると、指定されたドメインのリンクが IDN 上の他のすべてのドメインから解除されます。このときリンク解除の同期はとられません。このオプションは、指定したドメインから応答がまったくない場合、または AWOL 回復のために指定したドメインが IDN から分離されている場合に使用します。</p> <p>domain_name リンクを解除するドメインの名前を表します。指定できるドメインは 1 つの IDN に属するメンバーだけです。同時に複数のドメインのリンクを解除できますが、IDN 上の N-1 個 (N は、IDN 上の全ドメイン数) のドメインを指定すると、IDN 上のすべてのドメインのリンクが解除され、事実上、IDN が削除されることとなります。</p>
警告	<p>強制オプションは、ドメインの状態が不明 (AWOL) であることが分かっている場合の回復メカニズムとしてのみ使用してください。通常、このオプションは使用しないでください。このオプションを指定して、IDN の動作中にハードウェアを再プログラムすると、アービトレーション停止 (arbstop) が発生する可能性があります。IDN 上の複数のドメインがハングした場合には、ハングしたすべてのドメインのリンクを 1 回の操作で解除する必要があります。</p>

環境	IDN を作成する前、またはドメインを既存の IDN にリンクする前に、SUNW_HOSTNAME シェルの環境変数が IDN 機能をサポートするドメイン名に設定されていることを確認してください。この変数に関しては、プラットフォーム名では IDN コマンドは動作しません。
終了ステータス	domain_unlink が正常に完了した場合は、値 0 が返されます。正常に完了しなかった場合は、0 以外の値が返されます。
関連項目	domain_create(1M), domain_unlink(1M), domain_history(1M), domain_remove(1M), domain_rename(1M), domain_status(1M) Solaris Reference Manual (英語) の ifconfig(1M) 『Sun Enterprise 10000 InterDomain Networks ユーザーマニュアル』

名前	edd - イベント検出デーモン
形式	edd [-a] [-d]
注意	このコマンドは手動では実行しないでください。このコマンドは SSP スタートアップスクリプトによって自動的に呼び出され、必要に応じて再起動されます。
機能説明	<p>edd は Sun Enterprise 10000 の RAS 機能を実現するうえで核となるコンポーネントです。オプションを指定しない場合は、edd は edd.emc(4) 設定ファイルをチェックし、システムを制御する際に使用する検出スクリプトを判別します。チェック対象であるいずれかのイベントが検出されると、edd は edd.erc(4) 設定ファイルを使用して、処理を実行するかどうか、およびどの処理を実行するかを決定します。そして、指定された応答動作スクリプトがあれば、これを実行します。</p> <p>どの Sun Enterprise 10000 システムも、edd.emc(4) ファイルを 1 つ、グローバルイベント (システム全体に影響するイベント) 用の edd.erc(4) ファイルを 1 つ、各ドメイン用の edd.erc(4) ファイルを 1 つ備えています。これらのファイルの格納場所については、以下の「ファイル」を参照してください。監視対象となるイベントのリストについては edd.emc(4)、イベントに対して実行される処理のリストについては edd.erc(4) を参照してください。</p> <p>特定のイベントに対して edd が反応しないようにするには、edd.emc(4) ファイルを編集して、特定のイベント検出スクリプトのアップロードを無効にします。または、edd.erc(4) ファイルを編集して、特定の応答動作スクリプトを無効にします。</p> <p>edd で指定したオプションと、edd.emc(4) または edd.erc(4) で指定した処理が矛盾する場合は、前者が優先されます。</p> <p>購入先への注意事項： InterDomain Networks (IDN) に属するドメインのアービトレーション停止またはレコード停止が原因で edd によりダンプファイルが生成される場合は、該当する IDN 内のすべてのドメインの情報が出力されます。アービトレーション停止の場合は、edd により IDN ドメインのブートプロセッサの BBSRAM 情報も別ファイルに保存されます。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>-a 処理を実行しません。応答動作スクリプトは呼び出されません。</p> <p>-d アクティブな制御ボードにイベント監視スクリプトをアップロードしません。</p>

信号

SIGHUP イベント応答の設定ファイルを読み取ります。この信号は、以前に発生したイベントの応答動作には影響しません。

ファイル

以下のファイルがサポートされます。

`$SSPVAR/etc/platform_name/edd.emc`

`$SSPVAR/etc/platform_name/edd.erc`

`$SSPVAR/etc/platform_name/domain_name/edd.erc`

関連項目

`cbs(1M)`, `edd_cmd(1M)`, `hostview(1M)`, `snmpd(1M)`, `edd.emc(4)`, `edd.erc(4)`

名前	edd_cmd - edd へのコマンドの送信
形式	edd_cmd [-h] [-e] [-x <i>cmd</i>]
注意	このコマンドは主として購入先の技術者が使用します。このコマンドを使用する場合は、購入先に問い合わせてください。
機能説明	<p>edd_cmd はユーティリティーツールです。オプションを指定せずにこのコマンドを実行すると、最後に実行されたコマンドと現在の実行状態が表示されます。最後に実行されたコマンドは、start、stop、reconfig (rc 用)、または uninitialized (起動してから実行されたコマンドなし) のいずれかです。実行状態は、started-monitoring、stopped-monitoring、または uninitialized (起動の初期状態) のいずれかです。</p> <p>-h はヘルプオプションで、コマンド行オプションの概要を表示します。-e オプションを使用すると、システムのイベント監視機能の情報が表示され、プラットフォームとドメインに対するイベント監視機能が有効か無効かを示します。</p> <p>edd(IM) にコマンドを送るときに、edd_cmd は、edd(IM) がコマンドを受け取ったことの肯定応答を待つトラップハンドラを設定します。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>-h コマンド行オプションの概要を表示します。</p> <p>-e edd.emc および edd.erc ファイルを監視してイベントを表示します。</p> <p>-x <i>cmd</i> 指定したコマンドを実行します。<i>cmd</i> は以下のいずれかです。</p> <p> start - edd(IM) は、イベント検出の監視を開始します。</p> <p> stop - edd(IM) は、イベント検出の監視を停止します。</p> <p> rc - edd(IM) は、自身の設定ファイルを再度読み込んで、イベント検出の監視を実行します。</p>
使用例	<p>例 1 EDD の開始</p> <pre> % edd_cmd EDD: Command = uninitialized, State = stopped-monitoring % edd_cmd -x start % edd_cmd EDD: Command = start, State = started-monitoring </pre>

例 2 イベント監視情報の表示

```
% edd_cmd -e
EDD: Command = uninitialized, State = uninitialized

Event                Monitoring Platform Domain
                    edd.emc    edd.erc    edd.erc    Command
arbstop              enabled    none      enabled    Arbstopact
bulk_power_fail      enabled    enabled   none       BulkPowerFailact
bulk_power_norm      enabled    enabled   none       BulkPowerNormact
cb_power_off
```

ファイル

以下のファイルがサポートされます。

`$SSPETC/snmp/Ultra-Enterprise-10000.mib`

`$SSPVAR/etc/platform/edd.emc`

`$SSPVAR/etc/platform/edd.erc`

`$SSPVAR/etc/platform/domain/edd.emc`

関連項目

`edd(1M)`, `snmpd(1M)`, `straps(1M)`, `edd.emc(4)`, `edd.erc(4)`

名前	fad - ファイルアクセスデーモン
形式	fad
注意	このコマンドは手動では実行しないでください。このコマンドは SSP スタートアップスクリプトによって自動的に呼び出され、必要に応じて再起動されます。
機能説明	<p>fad デーモンは SSP 設定ファイルの監視、読み取り、書き込みを必要とする SSP クライアントのために、分散ファイルアクセスサービスを提供します。監視できるのは、fad_files(4) にリストされている読み取り可能ファイルだけです。</p> <p>fad は、lockf(3C) と同様のファイルロックサービスを提供します。あるクライアントによってファイルがいったんロックされると、同一ファイルに対する別のクライアントからの後続のロック要求はブロックされ、キューに入れられます。ファイルロックが解除されると、キューの先頭にあるクライアントに対するブロックは解除され、サービスが提供されます。</p> <p>fad は、machine_server(1M) などの他の SSP サーバーデーモンとともに動作します。SSP が一度に実行できる fad のインスタンスは 1 つだけです。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/.ssp_private/fad_files SSP 設定ファイルのリスト</p> <p>\$SSPVAR/pids/fad.pid プロセス ID ファイル</p>
関連項目	<p>machine_server(1M), fad_files(4)</p> <p>Solaris Reference Manuals (英語) の lockf(3C)</p>

名前	fan - ファンの電源と回転速度の表示または制御								
形式	<pre>fan fan -p on fan -p off [-l {front rear}] [-t FanTrayList] fan -s {nominal fast}</pre>								
機能説明	<p>オプションを指定しないで fan コマンドを実行すると、ファンの回転速度と電源の状態が表示されます。-s オプションまたは -p オプションを使用すれば、ファンの回転速度または電源の状態を変更できます。-p off オプションと -l および -t オプションを組み合わせると、電源を切断するファンを指定できます。fan -p on を使用すると、すべてのファンの電源を投入することができます。</p> <p>Sun Enterprise 10000 システムには、4 つのファンシェルフ (ファンの収納棚) が備わっています。そのうち 2 つはキャビネットの前面にあり、残りの 2 つは背面にあります。各シェルフにはそれぞれ 4 つのファントレイがあるため、合計で 16 個のファントレイがあることになります。1 つのファントレイには 2 つのファン (番号は 0 と 1) を収納することができます。したがって、システムあたり最大 32 個のファンを取り付けることができます。</p> <p>ファントレイは次のように配置されています。</p> <table border="0" data-bbox="602 1136 1406 1423"> <tr> <td>ファントレイ 0、1、2、3</td> <td>キャビネットの背面中段にあるシェルフ 0</td> </tr> <tr> <td>ファントレイ 4、5、6、7</td> <td>キャビネットの背面下段にあるシェルフ 1</td> </tr> <tr> <td>ファントレイ 8、9、10、11</td> <td>キャビネットの前面中段にあるシェルフ 2</td> </tr> <tr> <td>ファントレイ 12、13、14、15</td> <td>キャビネットの前面下段にあるシェルフ 3</td> </tr> </table> <p>各ファントレイは、同じ側の別のシェルフにあるファントレイと対になっています。一方のトレイが空気を吸い込んでキャビネットに送り、他方のトレイが空気を外に排出します。たとえば、シェルフ 0 のトレイ 3 はシェルフ 1 のトレイ 7 と対になっており、シェルフ 2 のトレイ 9 はシェルフ 3 のトレイ 13 と対になっています。</p>	ファントレイ 0、1、2、3	キャビネットの背面中段にあるシェルフ 0	ファントレイ 4、5、6、7	キャビネットの背面下段にあるシェルフ 1	ファントレイ 8、9、10、11	キャビネットの前面中段にあるシェルフ 2	ファントレイ 12、13、14、15	キャビネットの前面下段にあるシェルフ 3
ファントレイ 0、1、2、3	キャビネットの背面中段にあるシェルフ 0								
ファントレイ 4、5、6、7	キャビネットの背面下段にあるシェルフ 1								
ファントレイ 8、9、10、11	キャビネットの前面中段にあるシェルフ 2								
ファントレイ 12、13、14、15	キャビネットの前面下段にあるシェルフ 3								

オプション

以下のオプションをサポートしています。

- p on すべてのファンの電源を投入します。
- p off すべてのファンの電源を切断します。-1 および -t オプションの説明を参照してください。このオプションを -p off と一緒に使用すると、指定したファンの電源を切断することができます。-1 と -t が両方とも指定されている場合は、-1 オプションで指定されたファンが -t オプションで指定されたファンに優先されます。下の例を参照してください。
- 1 {front | rear} マシンの前面 (front) または背面 (rear) の指定したファンシェルフの電源だけを切断します。また該当する場合には、-t オプションで指定されたファントレーの電源が切断されます。このオプションは、power -p off オプションと一緒に使用したときだけ指定できます。
- t *FanTrayList* 指定したファントレーの電源を切断します。また状況に応じて -1 オプションによって指定されたシェルフの電源が切断されます。*FanTrayList* は、ファントレーの番号を 0 ~ 15 の整数で表したリストです (各ファントレーはスペースで区切られます)。このオプションは、power -p off オプションと一緒に使用したときだけ指定できます。
- s {nominal | fast} すべてのファンの回転速度を nominal (公称) または fast (高速) に設定します。どのファンも同じ速度で回転します。いったんファンの回転速度を変更すると、fan -s コマンドで回転速度を再度変更するまでその値が有効になります。デフォルトは nominal です。

使用例

例 1 ファンの回転速度を設定する

下記のコマンドは、ファンの回転速度をすべて高速に設定します。

```
fan -s fast
```

例 2 ファンの電源を切断する

下記のコマンドは、前面の 2 つのファンシェルフに収納されているすべてのファンの電源を切断します。このコマンドではファンシェルフ (-l) とファントレイリスト (-t) の両方が指定されていますが、優先されるのは前者です。

```
fan -p off -l front -t 0 1 2
```

関連項目

cbs(1M), edd(1M), hostinfo(1M), hostview(1M), snmpd(1M)

名前	fod - フェイルオーバーデーモン
形式	fod
注意事項	このプログラムは手動で実行しないでください。このプログラムは SSP 起動スクリプトによって呼び出され、必要に応じて監視・再起動されます。
機能説明	<p>fod デーモンは、常に Sun Enterprise 10000 プラットフォームの 2 つの SSP と制御ボードの状態を監視します。2 つの SSP で構成されている場合は、一方はメイン SSP として機能し、もう一方はメイン SSP に障害が発生した場合にバックアップとして機能します。同様に、一方の制御ボードは主制御ボードとして機能し、もう一方は主制御ボードに障害が発生した場合にバックアップとして機能します。</p> <p>fod は以下の接続を監視します。</p> <ul style="list-style-type: none"> ■ メイン SSP とスペア SSP 間 ■ Sun Enterprise 10000 ドメインを使用するメイン SSP とスペア SSP 間 ■ Sun Enterprise 10000 制御ボードを使用するメイン SSP とスペア SSP 間 <p>さらに、fod はディスク容量やメモリー使用率などの SSP の動作資源を監視します。メイン SSP の正常な動作を妨げる障害を検出すると、fod は障害のタイプに応じて SSP フェイルオーバーと制御ボードフェイルオーバーのいずれかを起動します。</p> <p>fod デーモンはメイン SSP とスペア SSP の両方で実行されます。どちらの SSP でも一度に 1 つのデーモンしか実行されません。両方の SSP の fod デーモンは、フェイルオーバープロトコル (FOP) による相互通信と Sun Enterprise 10000 制御ボード上で稼動する制御ボードエグゼクティブ (CBE) との通信を行います。FOP は TCP/IP の最上位に位置するアプリケーションレベルのプロトコルです。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/.ssp_private/ssp_resource SSP リソースファイル</p> <p>\$SSPVAR/.ssp_private/domain_config ドメイン設定ファイル</p> <p>\$SSPVAR/.ssp_private/cb_config 制御ボード設定ファイル</p> <p>\$SSPVAR/pids/fod-<i>platform_name</i>.pid プロセス ID ファイル</p> <p>\$SSPVAR/doors/fod Solaris ドアファイル</p>
関連項目	fad(1M)、datasyncd(1M)、cb_config(4)、domain_config(4)、ssp_resource(4)

名前	hostinfo - システム情報の表示
形式	hostinfo -F -S -h -p -t
機能説明	hostinfo はホスト関連のリアルタイムのデータを標準出力に出力します。hostinfo は snmpd(1M) デーモンにデータを問い合わせ、 snmpd(1M) は環境変数 SUNW_HOSTNAME に指定されている Sun Enterprise 10000 システムからこのデータを取り出します。
オプション	以下のオプションをサポートしています。 -F ファンの設定と状態を表示します。 -S 設定済みの各プロセッサについて、シグニチャーブロックの内容を表示します。 -h 設定済みの各プロセッサの状態とシグニチャーを表示します。 -p システムボード、制御ボード、ファントレー、センタープレーン、入出力キャビネットの電源装置をすべて表示します。 -t Sun Enterprise 10000 システムの周囲温度、および制御ボードとシステムボードの温度を表示します。温度はすべて摂氏で表示されます。
環境	環境変数 SUNW_HOSTNAME には、必ず、有効なドメインまたはプラットフォームの名前を設定してください。
ファイル	以下のファイルがサポートされます。 \$SSPETC/snmp/Ultra-Enterprise-10000.mib
関連項目	snmpd(1M)

名前	hostint - プロセッサへの割り込みと、カーネルのコアのダンプ
形式	hostint [-p <i>proc</i>] [-f] [-v]
機能説明	<p>hostint はハングしたドメインをパニックさせ、ダンプして、再起動します。-p オプションを指定しない場合は、hostint は環境変数 SUNW_HOSTNAME で指定されたドメインのブートプロセッサへ割り込み信号を送り、カーネルのコアをダンプします。hostint は、ブートプロセッサをパニックさせることに失敗すると、ドメイン内の他のプロセッサに割り込みを試みます。</p> <hr/> <p>注 - hostint は、ドメインがハングし、しかもそのハングが edd(1M) によって検出されなかった場合にだけ使用してください。通常は、edd(1M) がハングしたドメインを自動的に再起動します。-f オプションを使用しない限り、IDN ドメインをパニックさせることはできません。ただし、-f オプションはクラスタアービトレーション停止の原因となることがあります。</p> <hr/> <p>ドメインのハングに対応する際に、より手軽な別の方法をとることもできます。ドメインがハングした場合は、以下の手順を順番どおりに実行してください。</p> <ol style="list-style-type: none"> 1. 別のログオンでドメインを再起動します。これが成功した場合は、ドメインは完全にはハングしていなかったこととなります。コアはダンプされません。 2. コンソールからブレークを送信し、OpenBoot に割り込みます。これは、netcon(1M) または netcontool(1M) から実行することができます。OpenBoot に入ったら、sync コマンドを入力します。sync コマンドはパニックダンプと再起動を実行します。 3. hostint(1M) を実行します。これが成功した場合は、ドメインのパニック、ダンプ、再起動が実行されます。 4. hostreset を実行します。これが成功し、システムが自動的に起動しなかった場合は、ドメインを立ち上げます。 5. 上記のステップ 1 ~ 4 がすべて失敗した場合は、-f オプションを付けて bringup(1M) を実行します。

オプション

以下のオプションをサポートしています。

- p *proc* (ブートプロセッサではなく) 指定したプロセッサに割り込みシグナルを送ります。
- f IDN ドメインを強制的にパニックさせます。
- v 冗長 (Verbose)。標準出力に情報を出力します。

環境

環境変数 `SUNW_HOSTNAME` が該当ドメイン名に設定されている必要があります。

ファイル

以下のファイルがサポートされます。

`$SSPETC/snmp/Ultra-Enterprise-10000.mib`

関連項目

`bringup(1M)`, `netcon(1M)`, `snmpd(1M)`

名前	hostreset - ハングしたドメインのリセット
形式	hostreset
機能説明	<p>hostreset は環境変数 SUNW_HOSTNAME をリセットし、外部割り込み (XIR) を実行します。</p> <p>ドメインのハングに対応する際に、より手軽な別の手段をとることもできます。ドメインがハングした場合は、以下の手順を順番どおりに実行してください。</p> <ol style="list-style-type: none"> 1. 別のログインでドメインを再起動します。これが成功した場合は、ドメインは完全にはハングしていなかったこととなります。コアはダンプされません。 2. コンソールからブレークを送信し、OpenBoot に割り込みます。これは netcon(1M) または netcontool(1M) から実行することができます。OpenBoot に入ったら、sync コマンドを入力します。sync コマンドはパニックダンプと再起動を実行します。 3. hostint(1M) を実行します。これが成功した場合は、ドメインのパニック、ダンプ、再起動が実行されます。 4. hostreset(1M) を実行します。これが成功し、システムが自動的に起動しなかった場合は、ドメインを立ち上げます。 5. 上記のステップ 1 ~ 4 がすべて失敗した場合は、-f オプションを付けて bringup(1M) を実行します。
環境	環境変数 SUNW_HOSTNAME が該当ドメイン名に設定されている必要があります。
関連項目	bringup(1M), netcon(1M), edd(1M), hostint(1M)

名前	hostview - システムを監視するためのグラフィカルユーザーインタフェース (GUI)
形式	hostview
機能説明	<p>Hostview は、Sun Enterprise 10000 システムの設定・環境データを表示し変更するためのグラフィカルユーザーインタフェース (GUI) です。Hostview は SSP コマンドを生成します。</p> <p>Hostview を SSP ウィンドウのバックグラウンドで実行するには、次のコマンドを使用します。</p> <pre>hostview &</pre>
オプション	Hostview では、標準の X Toolkit のすべてのコマンド行オプションを利用できます。
環境	<p>別のディスプレイから Hostview を起動するには、環境変数 DISPLAY がそのディスプレイ用に設定されていて、環境変数 SUNW_HOSTNAME が該当プラットフォーム名またはドメインホスト名に設定されていることを確認してください。たとえば、次のようにして Hostview を実行します。</p> <pre>rlogin -l ssp ssp_host setenv DISPLAY local_host:0 hostview &</pre> <p>Hostview および Hostview の各種画面の詳細は、『Sun Enterprise 10000 SSP ユーザーマニュアル』を参照してください。</p>
関連項目	X11(7) Xlib Resource Manager のマニュアル

名前	hpost - JTAG 経由の POST の制御と処理
形式	<p>hpost [-? -h] [-?postrc -?blacklist -?level -?verbose]</p> <p>hpost [-aCcfngs] [-D [<i>boardmask</i>,] [<i>path</i>]] [-D "<i>comment</i>"] [-e <i>xcode</i>] [-g [<i>path</i> none]] [-H <i>boardmask,refproc</i>] [-i [<i>proc</i>]] [-J <i>bus_mask</i>] [-JJ <i>bus_mask</i>] [-l <i>level</i>] [-p <i>proc</i>] [-Q <i>proc</i> [,<i>skipmask</i>]] [-R{ <i>redlist_file</i> none}] [-v <i>level</i>] [-X{ <i>blacklist_file</i> none}] [-W [c]] [-Z <i>proc</i>]</p>
注意	このコマンドは購入先の技術者だけが使用します。このコマンドの使用を誤ると、オペレーティングシステムに重大な障害を引き起こす場合があります。
機能説明	<p>POST (電源投入時自己診断) プログラムは、Sun Enterprise 10000 システムの未初期化ハードウェアをチェックし、必要に応じて初期化済みシステムに設定したうえで、OBP (OpenBoot PROM) に渡します。hpost は SSP に常駐する実行可能プログラムであり、Sun Enterprise10000 ドメインと SSP の間の IEEE 1149.1 JTAG 走査インタフェースを介して、POST の動作とシーケンスを制御します。</p> <p>最初の引数が -n でない場合、hpost(1M) はオプションファイルの .postrc を読み取り、そのファイルに記述されている命令を実行してから、ホストとのやり取りを開始します (postrc(4) を参照)。hpost は、まず現在のディレクトリ (.) の中で .postrc を探します。ここで見つからない場合は、続いて \$SSPVAR/etc/<i>platform_name</i>/\$SUNW_HOSTNAME を探します。ここでも .postrc が見つからない場合は、ユーザーのホームディレクトリである \$HOME を探します。ただし例外として、カレントディレクトリが \$HOME の場合は、検索パスの最初の要素である (.) は省略されます。 .postrc が見つからない場合は、hpost(1M) は .postrc なしで処理を続行します。</p> <p>次のコマンドで表示できます。</p> <pre>hpost ?postrc</pre>
オプション	<p>hpost のオプションについて説明します。</p> <ul style="list-style-type: none"> ■ 数値オプションの引数は通常は 10 進数です。ただし、先頭に x または 0x を付けることによって、16 進数で指定することもできます。<i>mask</i> で示される引数は 16 進数になります。 ■ <i>board</i> はシステムボードの番号 (0 ~ 15) を示します。 ■ <i>proc</i> はプロセッサの番号 (0 ~ 63) を示します。この番号はプロセッサの物理的な位置 (例: <i>board</i> * 4 + <i>processor_module</i> #) に対応します。 ■ ハイフンの付いたフラグとその引数の間に、スペースは入りません。

- オプションの最初のグループの先頭にある疑問符 (?) は **h** で置き換えることができます。たとえば、`-?postrc` は `-hpostrc` で置き換えることができます。

<code>-?</code>	引数の簡単な説明を表示します。
<code>-h</code>	<code>-?</code> オプションの別名です。
<code>-?postrc</code>	<code>.postrc</code> ファイルの簡単な説明を表示します。
<code>-?blacklist</code>	<code>blacklist(4)</code> / <code>redlist(4)</code> ファイルの構文の簡単な説明を表示します。
<code>-?level</code>	<code>level</code> 番号に関する簡単な説明を表示します。
<code>-?verbose</code>	<code>verbose</code> 番号に関する簡単な説明を表示します。
<code>-a</code>	<code>POST</code> を代替診断レベルで実行します。 <code>-a</code> は、通常より高い診断レベルで <code>POST</code> を実行する必要があるエラー再起動シナリオで使用するために便宜的に用意されました。 <code>.postrc</code> ファイルで <code>alt_level</code> の値を指定した場合は、 <code>-a</code> はその値を代替診断レベルとして使用します。 <code>.postrc</code> ファイルで <code>alt_level</code> の値を指定しなかった場合は、デフォルト値を使用します。デフォルト値は、 <code>-a</code> を指定しないときのデフォルト診断レベルより高く設定されています。いずれにせよ、診断レベルが <code>-a</code> を指定しないときよりも低くなることはありません。 <code>-a</code> と <code>-l</code> を同時に使用することはできません。 <code>-l</code> および <code>postrc(4)</code> も参照してください。
<code>-C</code>	センタープレーンの初期設定を実行します。 <code>-C</code> を指定しないと、 <code>hpost</code> は、同じ物理プラットフォーム内にすでに設定済みであり実行中であるドメインがあると想定します。 <code>hpost</code> はセンタープレーンを調べて、バス設定を判別します。 <code>hpost</code> が調べるのはこの設定だけで、これが無効であると判別された場合は、 <code>hpost</code> はただちに終了します。 <code>-C</code> を指定した場合は、 <code>hpost</code> はこのプラットフォームでは他に何も実行されていないと想定し、ドメイン内で最高の性能指数を持つバス設定のシステム (センタープレーンも含まれます) をテストし、設定します。

- c** .postrc、blacklist(4)、redlist(4)の各ファイルを(通常通りに)構文解析します。構文エラーを検出した場合は、メッセージを表示してただちに終了します。postrc(4)、blacklist(4)、redlist(4)を参照してください。
- D[boardmask,][path]** 特別なモードのPOSTを起動します。この特別なモードでは、POSTはホストから状態を走査し、その結果をバイナリファイルにダンプして、ただちに終了します。pathが指定されていない場合は、デフォルトのパスが使用されます(boardmaskに続くコンマは必須です)。ダンプファイルはhpostでは分析できませんが、購入先とサンの担当者には有用なデータとなります。
boardmaskはマシンの20ビットマスクの部分をダンプに含めることを指定します。ビット0～15はシステムボード、ビット16と17はセンタープレーンの2つのハーフ、ビット18と19は2つの制御ボードにそれぞれ対応します。boardmaskを指定しないと、値はFFFFFFとみなされ、システムの全部分がダンプに含まれます。ただし、hpostは発見的決定アルゴリズムを使用して、接続されていないか電源が切れているシステムボードを検出し、ダンプファイルに含めないようにします。
boardmaskが指定されている場合、こうした編集は行われません。
-Dが指定されていると、hpostの標準入力端末である場合に、1行のコメントをファイルに入れることを促すメッセージが表示されます。Returnキーだけを押し、コメントは空白になります。一緒に-dを指定した場合は、コメントの入力を促すメッセージは表示されません。-dオプションの説明を参照してください。
- d "comment"** 指定したコメントをダンプファイルに挿入します。このオプションと引数の間にはスペースを入れます。単一の語からなるコメントの場合は、引用符は必要ありません。標準の入力手段(端末、スクリプト等)であるかどうかに関わらず、コメントが挿入されます。-Dを参照してください。

-excode	POST 終了コードのオンラインの説明を標準出力に出力します。
-g [path none]	画面のログファイルを作成します。ログファイルのデフォルトの名前は「ファイル」の項を参照してください。path には、デフォルトのログファイルを格納するディレクトリ、あるいは明示的なパスとファイル名のいずれかを指定します。none オプションを指定すると、.postrc ファイルでログファイルを開始した場合に、画面のログファイルの作成を抑制します。
-Hboardmask,refproc	特別なモードの POST を実行します。このモードでは、動作中のドメインに 1 つまたは複数のボードを DR (動的再構成) 接続するための準備が行われます。boardmask は、POST が実行されるボードの 16 ビットマスクで、refproc はターゲットドメインの現在のマスター CPU です。この CPU は別のボードに装着されている必要があります。 注意 - コマンド行では -H を指定しないでください。-H は DR プロセスの 1 ステップとして他のプログラムからのみ呼び出される必要があります。dr(1M) を参照してください。
-i[proc]	対話モードを実行します。このオプションはデバッグのために使用します。proc (単一プロセッサの番号) を指定すると、そのプロセッサだけが使用され、hpost は対話セッションを開始します。proc を指定しないと、すべてのプロセッサが使用され、hpost は、デバッグの各段階で、その段階を実行するか、次に進むかを尋ねるだけの単純な対話セッションを開始します。
-Jbus_mask	テストに基づいて設定を選択する代わりに、指定したバス設定を使用します。このモードの POST を使用するのにはサンの技術者だけあり、bus_mask は指定したバス設定における 6 ビットのバイナリマスクとして解釈されます。2 つの最上位ビットはデータバスであり、4 つの最下位ビットはアドレスバスです。3F はすべてのバスの使用を hpost に指示します。

<code>-JJbus_mask</code>	このオプションは <code>-J</code> オプションに似ていますが、実行されるのは JTAG の初期設定だけです。
<code>-llevel</code>	POST による診断レベルを設定します。有効値は 7 ~ 127 で、デフォルトのレベルは 16 です。 <code>-?level</code> を指定すると、簡単な説明が表示されます。
<code>-n</code>	<code>.postrc</code> ファイルの読み取りを抑止します。このオプションは必ず先頭に指定してください。
<code>-pproc</code>	指定したプロセッサを優先ブートプロセッサとして使用します。指定したプロセッサが最終設定に定義されていない場合は、このオプションは無視されます。

-Qproc[,skipmask]	<p>(Quick POST) このオプションを指定すると、POST は指定されたプロセッサのブートバス SRAM から設定情報を読み取り、最小限のテストを実行して、記述されたシステムを再設定します。ソフトウェアのクラッシュをすぐに回復する必要がある場合には、このオプションを使用します。指定した設定を実現できない場合は、このオプションは失敗します。skipmask オプションを指定すると、呼び出し側 (通常は別のプログラム) は、初期設定プロセスの中で不必要と判断したステップを省略します。このため、回復がより速くなります。skipmask のビットの意味を以下に示します。</p> <p>0 - 再設定の全段階を実行します (デフォルト)。</p> <p>0x0001 - プロセッサ IMU のタグをクリアせず、IMU を無効にしません。</p> <p>0x0002 - プロセッサ DMU のタグをクリアせず、DMU を無効にしません。</p> <p>0x0004 - プロセッサの命令キャッシュをクリアしません。</p> <p>0x0008 - プロセッサのデータキャッシュをクリアしません。</p> <p>0x0010 - プロセッサの外部キャッシュをクリアしません。</p> <p>0x0020 - CIC 重複タグ (DTAG) をクリアしません。</p> <p>0x0040 - 入出力コントローラを初期設定しません。</p> <p>0x0080 - メモリーをクリアしません。</p>
-q	省略 (Quiet) モード。画面出力をすべて抑止します。
-R {redlist_file none}	デフォルトの redlist ファイルの代わりに使用する redlist を指定します。none を指定すると、redlist ファイルは使用されません。このセクションの最後に記述されている「注意」を参照してください。
-s	適切な優先順位を付けたうえで、画面出力をすべて syslog に送ります。
-vlevel	メッセージの冗長レベルを設定します。有効値は 0 ~ 255 で、デフォルトのレベルは 20 です。-?verbose を指定すると、簡単な説明が表示されます。

- X** {*blacklist_file* | none} デフォルトのパスは使用しません。none を指定すると、**blacklist(4)** ファイルは使用されません。このセクションの最後に記述されている「注意」を参照してください。
- w** [c] 現在のドメインに存在しているレコード停止 (**Recordstop**) 状態をクリアし、センタープレーンの ASIC レコーディングを再度有効にしようとしています。このドメインにアービトレーション停止やその他の重大なエラーが検出されたときは、報告したうえで、レコード停止をクリアしようとする試みは中止されます。この操作の結果は、hpost の終了コードで報告されます。
レコード停止状態が別のドメインに存在している場合は、センタープレーンのレコーディングを再度有効にする試みは失敗します。この失敗は報告されますが、-w の操作は失敗したとはみなされません。
InterDomain Networks (IDN) に属するドメインに対する -w 操作では、オプションの c フラグを使用してください。IDN の 1 つのドメインに対してのみ -wc を呼び出してください。hpost は、そのドメインおよび IDN 内の他のすべてのドメインのレコード停止をクリアします。
通常、このモードの hpost は、状態ダンプファイルが作成されてから、SSP イベント検出デーモンによって呼び出されます。-D オプションを参照してください。
- Zproc** (**Zip POST**) このオプションを指定すると、POST は指定したプロセッサのブートバス SRAM から設定情報を読み取り、システムの JTAG 初期設定だけを実行して、記述されている設定を実現します。ソフトウェアのクラッシュの後でソフトウェアの状態をダンプする場合は、このオプションを使用します。通常の POST プロセスではこの状態は破壊されるおそれがあります。指定した設定を実現できない場合には、hpost -Z は失敗したものとみなされます。

注意 – プロダクションシステムで -R または -X を使用するときには注意が必要です。POST が標準外の **blacklist(4)** ファイルまたは **redlist(4)** ファイルを使用していることは、他の SSP ソフトウェアにはわかりません。

使用法	通常、POST は SSP の監視スクリプトまたは監視プログラムによって実行されます。ただし、エンジニアリング開発、製造、保守サービスなどの状況では、コマンド行から実行されることもあります。
ファイル	<p>以下のファイルがサポートされます。</p> <pre> ./ .postrc ローカルの POST 設定ファイル \$HOME/ .postrc ユーザーのデフォルト POST 設定ファイル \$SSPVAR/etc/platform_name/\$SUNW_HOSTNAME/ .postrc ホスト名に固有のデフォルト POST 設定ファイル \$SSPVAR/etc/platform_name/blacklist デフォルトの blacklist ファイル (-X を参照) \$SSPVAR/etc/platform_name/redlist デフォルトの redlist ファイル (-R を参照) \$SSPVAR/adm/\$SUNW_HOSTNAME/post/postmdd.hhmm.log デフォルトのログファイル (-g を参照) \$SSPVAR/adm/\$SUNW_HOSTNAME/xfstatemdd.hhmm.ss デフォルトのダンプファイル (-D を参照) \$SSPVAR/adm/\$SUNW_HOSTNAME/hpost.lock POST ロックファイル \$SSPOPT/release/Ultra-Enterprise-10000/*/*/*/* hostobjs/*.elf (ホストに常駐している) POST の実行可能ファイルをダウンロードするときのパス </pre>
環境	環境変数 SUNW_HOSTNAME が該当ドメイン名に設定されている必要があります。
診断	0 ~ 63 の終了ステータスは、正常な設定を意味します。この数値は、POST 対 OBP のハンドオフ構造体 (設定を記述します) を持つブートバス SRAM が含まれているプロセッサの数を示します。0 ~ 63 以外の数値は、システムが設定されていないことを示します (これらはサン

術者だけが使用する数値であり、関連情報は使用制限の見出しが付いたファイルに記述されています。ただし、`-e` オプションを指定すると、すべての値の説明が表示されます)。

関連項目

`obp_helper(1M)`, `redx(1M)`, `blacklist(4)`, `postrc(4)`, `redlist(4)`

名前	initcmdsycn, cmdsycn, cancelcmdsycn, savecmdsycn - コマンド同期コマンド
形式	<p>cancelcmdsycn <i>cmdsycn_descriptor</i></p> <p>initcmdsycn <i>script_name</i> [parameters]</p> <p>savecmdsycn -M <i>identifier</i> <i>cmdsycn_descriptor</i></p>
機能説明	<p>一連のコマンド同期コマンドは、連動して SSP フェイルオーバーによって割り込まれたユーザー定義のスキ립トの回復を制御します。コマンドの同期を有効にするには、ユーザー定義のスキ립トに以下のコマンドを挿入します。</p> <ul style="list-style-type: none"> ■ initcmdsycn は、回復するスキ립トを識別するためのコマンド同期記述子を生成します。 <p>この記述子は、コマンド同期リストに追加されます。この記述子によって、フェイルオーバーの後に新しいメイン SSP で再起動するスキ립トとコマンドを識別します。</p> <ul style="list-style-type: none"> ■ savecmdsycn は、フェイルオーバーの後に処理を再開できるスキ립ト上の場所を特定するマーカーを追加します。 ■ cancelcmdsycn は、コマンド同期リストからコマンド同期記述子を削除します。削除することで、確実にスキ립トは一度だけ実行され、次のフェイルオーバーの後には実行されません。 <p>スキ립トのすべての終了パスに cancelcmdsycn シーケンスを記述して、コマンド同期リストから記述子を確実に削除します。記述子を削除しない状態でフェイルオーバーが起動すると、スキ립トは新しいメイン SSP で再実行されます。</p> <hr/> <p>注 - コマンドの同期を有効にするには、initcmdsycn と cancelcmdsycn の両方のシーケンスをスキ립トに記述する必要があります。 savecmdsycn コマンドは省略可能ですが、処理を再開できるスキ립ト上の特定の場所を指定する場合のみ使用します。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p><i>cmdsycn_descriptor</i> ユーザー定義のスキ립トを識別するコマンド同期記述子を指定します。この記述子は、initcmdsycn コマンドによって返される標準出力値です。</p> <p>-M <i>identifier</i> フェイルオーバーの後に再開できるスキ립ト上の場所を指定します。この識別子には、必ず正の整数を使用します。</p>

拡張機能説明

parameters ユーザー定義のスクリプトに関連するオプションとパラメタを指定します。これらのパラメタはスペア SSP に格納され、フェイルオーバーの後に指定されたスクリプトを再起動するときに使用されます。

script_name 同期させるユーザー定義のスクリプトの名前を指定します。

コマンド同期コマンドは、ユーザー定義スクリプトにおける決められた論理上の位置に挿入します。

たとえば Korn シェルスクリプトの場合は、以下のような構造になります。

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any interrupts.
# Use the cancelcmdsycn to remove the script from the command
# synchronization list. Otherwise, the command will get restarted
# on the new main SSP.
#
clean_up () {
    cancelcmdsycn $desc
    exit
}
# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point if provided
#
for arg in $*; do
    case $arg in
        -M )
            goto_label=$arg;;
        .
        .
        .
    esac
done
# Place this script and all its parameters in the command synchronization
# list, which indicates the commands to be restarted after an SSP failover.
#
# NOTE: The script must be executable by user ssp and reside in the same
# directory on both the main and the spare SSP. If the command is not
# part of the defined PATH for user ssp, the absolute filename must be
# passed with the initcmdsycn command.
#
initcmdsycn script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been processed or an
# error is detected.
#
while (( $goto_label != 0 )) ; do
    # Each case should represent a synchronization point in the script.
    #
    case $goto_label in
        #
        # Step 1: Do something
        #
        1 )
            do_something
            .
            .
            .
            # Execute the savecmdsycn command with the script's
            # descriptor and a unique marker to save the position.
            # If a failover occurs here, the commands
            # represented in the next goto_label (2) will be
```

```

        # resumed.
        #
        savecmdsycn -M $(( $goto_label + 1 )) $desc
        goto_label=$(( $goto_label + 1 ))
        ;;
#
# Step 2: Do more things
#
2 )
    do_more_things
    .
    .
    savecmdsycn -M $(( $goto_label + 1 )) $desc
    goto_label=$(( $goto_label + 1 ))
    ;;
#
# Step 3: Finish the last step and set the goto_label to 0
# so that the script ends.
3 )
    finish_last_step
    .
    .
    goto_label=0
    ;;
esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsycn to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsycn $desc

```

終了ステータス

以下の終了ステータスが返されます。

0	正常に終了しました。
0 以外	エラーが発生しました。

注 - `initcmdsycn` に対する標準出力は、コマンド同期記述子で構成されます。また、フェイルオーバーが起動しない場合 (フェイルオーバーの後、またはシングル SSP 環境の場合) は、同期コマンドを含むスクリプトによってプラットフォームログファイルにエラーメッセージが生成され、0 以外の終了コードが返されます。このエラーメッセージは、無視しても問題はありません。

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

関連項目

runcmdsync(1M), showcmdsync(1M)

ディレクトリ /opt/SUNWssp/examples/cmdsync の (同期コマンドを含む) ユーザー定義スクリプトの例

名前	machine_server - マシンサーバー
形式	machine_server [-m]
注意	このデーモンは手動では実行しないでください。このデーモンは SSP スタートアップスクリプトによって自動的に呼び出され、必要に応じて再起動されます。
機能説明	<p>machine_server デーモンは以下の機能を実行します。</p> <ul style="list-style-type: none"> ■ netcon_server(1M) からの TCP ポート登録要求に応え、Sun Enterprise 10000 SNMP エージェント (snmpd(1M)) からの UDP ポート登録要求に応えます。 ■ 各種の SSP クライアントプログラムからの netcon_server(1M) ポートと snmpd(1M) ポートのルックアップ要求に応え、ポート番号を返します。 ■ エラーメッセージを適切なメッセージファイルに送ります。 <p>SSP が一度に実行できる machine_server のインスタンスは 1 つだけです。</p>
オプション	<p>以下のオプションをサポートしています。</p> <p>-m machine_server デーモンがメイン SSP 上で動作していることを示します。このオプションを省略した場合は、machine_server デーモンがスベア SSP 上で動作しているものとみなされます。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/pid/machine_server.pid プロセス ID ファイル</p> <p>/etc/services 予約されたポート指定</p>
関連項目	netcon_server(1M), snmpd(1M)

名前	netcon - ネットワークコンソール
形式	netcon [-d -f -g -l -r] [-e <i>esc_char</i>]
機能説明	<p>netcon コマンドは、ドメインのホストコンソールプログラムに対するリモート接続を実現し、SSP ウィンドウ (このウィンドウでコマンドが実行されます) を環境変数 <code>SUNW_HOSTNAME</code> により指定されたドメインのコンソールウィンドウにします。netcontool(1M) も参照してください。netcontool(1M) は netcon を実行するメニュー方式のプログラムです。</p> <p>ネットワーク上の複数のシステムで、複数のドメインコンソールウィンドウ (「コンソールセッション」とも呼ばれます) を同時に開くことができます。各コンソールウィンドウはそれぞれのドメインホストの出力を表示します。ただし、特定のドメインについて、複数のコンソールウィンドウが同時に書き込み権限を持つことはできません。書き込み権限には「ロックなし」のものと「ロック付き」のものがあります。</p> <p>ロックなしの書き込み権限の保護レベルはあまり高くありません。netcon -g、netcon -l、または netcon -f によって別のコンソールセッションが始まった場合、あるいは別のドメインコンソールウィンドウで <code>~@</code>、<code>~&</code>、または <code>~*</code> が実行された場合には、既存のロックなしの書き込み権限は消失します。チルド (~) コマンドについては、下記の「使用法」を参照してください。</p> <p>ロック付きの書き込み権限の保護レベルはもっと高くなります。この権限が消失するのは、netcon -f によって別のコンソールセッションが始まった場合、あるいは別のドメインコンソールウィンドウで <code>~*</code> が実行された場合だけです。いずれの場合も、新しいコンソールセッションは排他セッションであり、他のセッションはすべて終了します。</p> <p>該当するドメインの他のコンソールセッションに書き込み権限がないときに、オプションを指定せずに netcon を実行すると、ロックなしの書き込み権限付きのドメインコンソールウィンドウが始まります。他のセッションにすでに書き込み権限が与えられている場合には、読み取り専用モードのセッションが始まります。</p> <p>netcon では、ドメインコンソールウィンドウとの通信には、標準のネットワークインタフェースまたは JTAG インタフェースのいずれかを使用できます。これらのインタフェースは、チルドイコール (~=) コマンドを使用すると、手動で切り替えることができます。</p>

オプション

同時に複数のオプションを付けて `netcon` を実行することはできません。ただし、例外は、`-e` オプションで他のどのオプションとも組み合わせ使用できます。複数のオプションを指定すると、最も優先順位の高いオプションだけが有効になります。オプションは `-f`、`-l`、`-g`、`-r` の順に優先されます。

- `-d` デバッグモードをオンにします。このオプションを使用する前に、必ずサービスアドバイザーに相談してください。
- `-e esc_char` デフォルトのエスケープ文字 (`~`) を *esc_char* に変更します。*esc_char* には、次の文字以外であればどんな文字でも指定できます。`@`、`#`、`^`、`&`、`*`、`(`、`)`、`=`、`|`、`[`、`:`、`"`、`'`、`<`、`>`、`..`、`?`
- `-f` (**Force** オプション) ロック付きの書き込み権限でドメインコンソールウィンドウを開き、他のすべてのコンソールセッションを終了して、新しいセッションを開始できないようにします。このオプションは排他セッションモードを開始します。コンソールを排他的に使用する必要があるとき(たとえば、プライベートのデバッグセッションを実行するとき)には、このオプションを選択してください。
注 - マルチセッションのモードに復帰するには、(`~^` を実行して) ロックを解除するか、または (`~.` を実行して) **netcon -f** セッションを終了します。
- `-g` (**Grab** オプション) ロックなしの書き込み権限でドメインコンソールウィンドウを開きます。別のセッションがロックなしの書き込み権限を与えられている場合、新しいドメインコンソールウィンドウがこの権限を奪うこととなります。別のセッションがロック付き書き込み権限を与えられている場合、この要求は拒否され、読み取り専用のセッションが始まります。
- `-l` (**Lock** オプション) ロック付きの書き込み権限でコンソールウィンドウを開きます。別のセッションがロックなしの書き込み権限を与えられている場合、新しいドメインコンソールウィンドウがこの権限を奪うこととなります。別のセッションがロック付き書き込み権限を与えられている場合、この要求は拒否され、読み取り専用のセッションが始まります。
- `-r` (**Read Only** オプション) 読み取り専用権限でドメインコンソールウィンドウを開きます。

使用法

ドメインコンソールウィンドウでは、行の最初に表示されるチルダ (~) は、エスケープシグナルとして解釈され、netcon に対して次のような特別な動作をとるように指示します。

- ~# OBP または kadb に割り込みます。
- ~. 接続を解除して、netcon セッションを終了します。
- ~@ ロックなしの書き込み権限を取得します (-g を参照)。
- ~^ 書き込み権限を解除します。
- ~? 実行中の全コンソールセッションおよび使用中の通信経路の状態を表示します。
- ~= ネットワークと JTAG インタフェースの通信経路を切り替えます。~= は、(~* を実行して) プライベートモードでのみ使用してください。
- ~& ロック付きの書き込み権限を取得します (-l を参照)。このシグナルは、読み取り専用セッションあるいはロックなしの書き込み権限によるセッションが実行されているときに出すことができます。
- ~* ロック付きの書き込み権限を取得し、他のセッションをすべて終了して、新しいセッションを開始できないようにします (-f を参照)。マルチセッションのモードに復帰するには、ロックを解除するか、またはこのセッションを終了します。

注意事項

ドメインコンソールウィンドウで vi(1) を実行し、エスケープシーケンス (チルダコマンド) を正常に実行するには、環境変数 TERM と Netcon ウィンドウの設定が一致してはなりません。たとえば、netcon コマンドを実行したウィンドウで TERM が xterm に設定されている場合は、ドメインコンソールウィンドウの TERM も xterm に設定されている必要があります。

該当ドメインで bringup(1M) が実行されていない場合には、以下のエラーメッセージが表示されます。

```
netcon_server is not running for domain_name.
```

このエラーメッセージが表示された場合は、netcon_server -r を実行して netcon セッションを再起動します。再起動できない場合は bringup(1M) を実行します。

netcon セッションのログへの記録はデフォルトで有効になっており、出力は /var/opt/SUNWssp/adm/domain_name/netcon ファイルに記録されます。

netcon コンソールセッションを **kill -9** コマンドで終了させた場合、netcon コマンドが実行されていたウィンドウまたは端末が raw モードとなり、ハングした状態になります。この状態を解消するには、**^j** と入力し、次に **stty sane** と入力し、さらに **^j** と入力します。

netcon セッションをネットワーク通信モードに切り替えることができない場合は、次の操作を実行します。

1. メイン SSP から、**rlogin** コマンドを使用してドメインにスーパーユーザーとしてログインします。
2. 次のように入力して **cvcd** を再起動します。

```
# /etc/init.d/cvc stop  
# /etc/init.d/cvc start
```

環境

環境変数 **SUNW_HOSTNAME** が該当ドメイン名に設定されている必要があります。

関連項目

netcontool(1M)

SunOS リファレンスマニュアルの **boot(1M)**、**cvcd(1M)**。

名前	netcon_server - ネットワークコンソールのサーバーデーモン
形式	<pre>netcon_server [-S] -p boot_proc</pre> <pre>netcon_server [-S] -r</pre>
注意	このデーモンは、bringup(1M) によって実行されます。コマンド行では実行しないでください。
機能説明	<p>netcon_server は、各種の SSP コンソールウィンドウ（「netcon 接続セッション」とも呼ばれます）と、(SSP ウィンドウの環境変数 SUNW_HOSTNAME で指定された) ドメインのブートプロセッサとの間のやり取りを管理します。</p> <p>ドメインが立ち上がっているときは、netcon_server は、各種のドメインコンソールウィンドウと、ドメイン側で動作している cvcd(1M) デーモンとの間の中継として機能します。ドメインがダウンしているときは、netcon_server は、ドメインコンソールウィンドウと OBP との間の中継として機能します。</p> <p>netcon_server はデーモンのように動作します。つまり、自分自身のコピーを生成して親プロセスを終了し、制御している端末から自分自身を効果的に接続解除します。</p>
オプション	<p>以下のオプションをサポートしています。</p> <pre>-S</pre> <p>/etc/syslog.conf で syslogd の local1 機能が有効にされている場合は、netcon session のログの記録を中断します。</p> <pre>-p boot_proc</pre> <p>指定されたブートプロセッサを使用します。</p> <pre>-r</pre> <p>netcon_server を再起動し、snmpd(1M) デーモンからドメインブートプロセッサを取得します。 netcon_server -r は、netcon_server が停止し、ホストが立ち上がっているか OBP プロンプトの場合にのみ実行してください。</p>
環境	環境変数 SUNW_HOSTNAME が該当ドメイン名に設定されている必要があります。
関連項目	<p>『Sun Enterprise 10000 SSP リファレンスマニュアル』の bringup(1M), ssp_startup(1M), netcon(1M), thermal_config(1M)。</p> <p>SunOS リファレンスマニュアルの cvcd(1M)</p>

名前	netcontool - ネットワークコンソールのツール
形式	netcontool
機能説明	<p>netcontool は netcon(1M) に対するグラフィカルユーザーインタフェース (GUI) です。netcontool は、SSP の環境変数 SUNW_HOSTNAME で指定されたドメインへのコンソール接続に必要な設定情報を受け入れ、その接続を実現します。この結果、ユーザーは netcon プログラムが受け入れるエスケープシーケンスと同様の機能を、ボタンをクリックして実行できるようになります。エスケープシーケンスについては、netcon(1M) を参照してください。</p> <p>netcontool コマンドを実行すると、ドメインコンソールウィンドウで以下のボタンを使用できるようになります。</p> <p>Exit netcontool を終了します。</p> <p>Configure コンソール設定のダイアログボックスを表示します (下記を参照)。</p> <p>Connect netcon セッションを開始します。接続するのにもっと詳細な設定情報が必要な場合は、そのことを知らせるメッセージが表示されます。このメッセージが表示された場合は、Configure ボタンをクリックしてください。</p> <p>Configure ボタンをクリックすると、Console Configuration ダイアログボックスが表示され、以下の情報の入力が可能になります。</p> <p>端末の種類 適切なボタンをクリックして端末の種類を選択します。ドメインコンソールウィンドウで vi(1) を実行し、エスケープシーケンス (チルダコマンド) を正常に実行するには、ここで指定する端末の種類がドメインコンソールウィンドウの環境変数 TERM と一致している必要があります。たとえば、ドメインコンソールウィンドウの TERM が xterm の場合は、ここでも xterm (Xterm ボタン) を指定する必要があります。デフォルトは Xterm です。</p> <p>セッションの種類 Read Only Session、Unlocked Write (netcon -g)、Locked Write (netcon -l)、または Exclusive Session (netcon -f) のボタンをクリックしてください。</p>

デフォルトの設定は **Read Only** (読み取り専用) です。ただし、セッションが実際に読み取り専用で開始されるのは、すでに実行されているセッションが少なくとも 1 つある場合です。他に実行されているセッションがない場合は、**Read Only** ボタンを選択しても、新しいセッションはロックなしの書き込みモードで始まります。

Done (終了) ボタンをクリックすると、**Connect** ボタンが選択可能になります。**Connect** ボタンをクリックすると、新しいセッションが始まります。設定情報が不足している場合は、そのことを知らせるエラーメッセージが表示されます。

Connect をクリックすると、netcontool が netcon(1M) プログラムを起動し、指定した端末の種類のコマンドラインウィンドウが現れます。このウィンドウでは、**Configure** と **Connect** を除くすべてのボタンが選択可能になっています。各ボタンは、マニュアルページの netcon(1M) に説明されているエスケープシーケンスに対応しています。netcontool を使用している間は、キーボードからエスケープシーケンスを入力することはできません。

ウィンドウ管理の操作の間は、コマンドラインウィンドウと netcontool ウィンドウは別々に機能します。

環境

環境変数 **SUNW_HOSTNAME** が該当ドメイン名に設定されている必要があります。

関連項目

netcon(1M)

名前	obp_helper - システムメモリーへの OpenBoot のダウンロード
形式	obp_helper [-eivqr] [-o <i>Filename</i>] [-d <i>Filename</i>] [-m <i>boot_proc</i>] [-A {on off}] [-D {on off}] [<i>boot-arguments</i>]
機能説明	<p>注 - 通常、obp_helper はコマンド行からではなく、bringup(1M) スクリプトによって実行されます。コマンド行からは、obp_helper -e として実行できる以外に、特定の状況では obp_helper -r としても実行できます。</p> <hr/> <p>obp_helper は OpenBoot を実行する環境を用意します。続いて、OpenBoot をダウンロードし、時刻サービスと EEPROM シミュレーションサービスを提供します。</p> <p>obp_helper は OpenBoot が終了するまで、または bringup(1M) が実行されるまで実行されます。</p> <p>obp_helper はデーモンのように動作します。つまり、自分自身のコピーを生成して親プロセスを終了し、制御している端末から自分自身を効果的に接続解除します。</p>
オプション	<p>以下のオプションをサポートしています。</p> <ul style="list-style-type: none"> -e obp_helper によって変更可能な EEPROM オプションの現在の設定を表示します。 -i EEPROM の起動引数を無効にします。 -v 冗長 (verbose) モードで実行します。 -q 省略 (quiet) モードで実行します。 -r obp_helper を再起動します。obp_helper -r を実行するのは、OpenBoot がすでに実行されていて、obp_helper が停止または消失した場合だけに限ってください。 -o <i>filename</i> 標準の OpenBoot 実行可能ファイルの代わりに、指定したファイルをダウンロードします。 -d <i>filename</i> 標準の download_helper 実行可能ファイルの代わりに、指定したファイルをダウンロードします。 -m <i>boot_proc</i> <i>boot_proc</i> によって指定したプロセッサをブートプロセッサとして使用します。

- A {on | off}** OpenBoot の自動起動モードを有効 (on) または無効 (off) にします。OpenBoot に対するこのオプションの効果は、`setenv auto-boot? [true|false]` コマンドと同じです。このオプションによって、OpenBoot をシミュレートした EEPROM 内の `auto-boot?` フラグの状態が変更されます。`auto-boot` が有効な場合には、`bringup(1M)` コマンドは、他の OBP 変数によって決定された適切な起動ディスクを使用してシステムを完全に立ち上げます。`auto-boot` が無効な場合には、`bringup(1M)` コマンドは、`netcon(1M)` ウィンドウに OBP プロンプト (OK) を表示して終了します。
- D {on | off}** OpenBoot の診断モードを有効 (on) または無効 (off) にします。OpenBoot に対するこのオプションの効果は、`setenv diag-switch? [true|false]` コマンドと同じです。このオプションによって、OpenBoot をシミュレートした EEPROM 内の `diag-switch?` フラグの状態が変更されます。
- boot-arguments* 起動引数を OpenBoot の起動コマンドに 1 つずつ渡します。`disk` や `net` などの OpenBoot の標準デバイスのエイリアスを指定できます。これらの引数は現在の起動にのみ影響します。これに続く起動コマンドは、OpenBoot の標準引数である `boot-device` と `boot-file` を使用します。

環境

環境変数 `SUNW_HOSTNAME` が該当するドメイン名に設定されている必要があります。

関連項目

`bringup(1M)`

SunOS リファレンスマニュアルの `boot(1M)`

名前	power - 電源の制御
形式	<pre> power power [-sb <i>list</i>] [-cb 0 1] [-csb <i>csb_list</i>] [-p <i>p_list</i>] power -off [-f] [-all] [-sb <i>sb_list</i>] [-csb <i>csb_list</i>] [-p <i>p_list</i>] [-cb 0 1] [-d -q] power -off -B [-f] [-d -q] power -off -ps <i>ps_list</i> [-d -q] (購入先の技術者専用) power -on [-all] [-sb <i>sb_list</i>] [-csb <i>csb_list</i>] [-p <i>p_list</i>] [-d -q] power -m power -m <i>margin_list</i> -sb <i>sb_list</i> [-d -q] (購入先の技術者専用) power -m <i>margin_list</i> -s -sb <i>sb_list</i> [-d -q] (購入先の技術者専用) power -m <i>margin_list</i> [-d -q] (購入先の技術者専用) power -m <i>margin_list</i> -s [-d -q] (購入先の技術者専用) power -v [-all] [-sb <i>sb_list</i>] [-csb <i>csb_list</i>] [-p <i>p_list</i>] [-ps <i>ps_list</i>] [-d -q] </pre>
機能説明	<p>power コマンドは Sun Enterprise 10000 システムの電源と入出力キャビネットの状態を表示し制御します。このコマンドを使用して、個々のボードや入出力キャビネットの電源を投入または切断したり、過電圧 / 過小電圧の監視に使用する電圧の限界値を表示または設定することができます。</p> <p>オプションを指定せずに power コマンドを実行すると、48 ボルト電源、入出力電源、制御ボードとセンタープレーンサポートボードの電源、個々のシステムボードの電源の状態がすべて表示されます。下記の「使用例」を参照してください。</p>

センタープレーンサポートボードとシステムボードの電源の状態は、次のように表示されます。

```

Good 48V Bulk Power Supplies:                0 2 3 4
Number of Good 48V Bulk Power Supplies:      4 (no redundancy)
Required 48V Power Supplies for 8 System Boards: 4
Number of Good Peripheral Cabinet Power Supplies: 0
Centerplane Support Board Average Voltages (V):
CSB#      5VDC Vcc HK      3.3VDC Vdd HK      3.3VDC Vdd Core
-----
0         5.047      5.037          3.501          3.311      3.301      3.297
1         5.017      5.017          3.491          3.303      3.302      3.302

System Board Average Voltages (V):
SB#      3.3VDC      5VDC      3.3VDC      VDC      5VDC
        Vdd      Vcc HK      Vdd HK      Vdd Core      Vcc
-----
0         3.301      5.086      3.407          2.601      5.008
1         3.300      5.022      3.423          2.601      5.013
2         3.300      5.015      3.439          2.607      5.010
3         3.301      5.035      3.434          2.599      5.013
4         3.290      5.015      3.423          1.897      5.005
7         3.301      5.052      3.418          2.604      5.003
8         3.296      5.059      3.429          2.596      5.000
9         3.302      5.025      3.412          2.594      5.008

Control Board Average Voltages (V):
CB#      5VDC      5VDC      3.3VDC      5VDC Vcc      5VDC
        Vcc      Vcc HK      Vdd HK      Peripheral      Vcc Fans
-----
0         5.113          5.049          3.420          5.194          5.091

```

上記で、

- 最初のカラムの 5.0VDC と 3.3VDC は各電源の入力電圧値です。VDC は直流電圧 (Voltage Direct Current) を意味します。センタープレーンのリストでは、5.0VDC HK が 2 回、3.3VDC Vdd が 3 回繰り返されています。これは、5.0VDC HK が 2 つの異なる位置で測定され、3.3VDC Vdd が 3 つの異なる位置で測定されたことを意味します。

システムボードのリストでは、Vdd Core の値は装着されているプロセッサの種類に依存します。この値の範囲は 2.5 ~ 3.7VDC です。JTAG はアナログ/デジタル変換器にアクセスして、プロセッサモジュール上の抵抗器を読み、実際の値を判別します。

- HK はハウスキューピング電力を意味します。48 ボルト電力がボードに供給されているときには、必ずこの電力は供給されています。
- Vdd は +3.3 VDC を意味します。
- Vcc は +5.0 VDC を意味します。
- 2 番目のカラムは、JTAG がアクセスしたアナログ/デジタル変換器が示す実際の電圧値を示しています。

注 - ハウスキーピング電力がマシンに供給されると、自動的に制御ボードの電源が入ります。システムボードとセンタープレーンサポートボードの電源は、`power` コマンドを実行して投入します。InterDomain Networks (IDN) のメンバーであるドメインのシステムボードの電源を切断することはできません。電源を切断するには、まず `domain_unlink(1M)` を使用してそのドメインのリンクを解除する必要があります。IDN に属するボードの電源を強制的に切断する場合には、`power -f` コマンドを実行してください。ただし、このコマンドを実行するとクラスタがアービトレーション停止してしまふことがあります。IDN がある場合には、Sun Enterprise 10000 のすべてのホストキャビネットの電源を切断することはできません。

オプション

オプションの有効な組み合わせについては、上記の「形式」を参照してください。

`-sb sb_list` 指定したシステムボードの電源を投入する、切断する、あるいはその電源装置への電力供給を有効にします。`sb_list` はスペースで区切られた 0 ~ 15 の整数のリストであり、システムボードの番号を示します。このオプションは限界値の設定の際にも使用します。`-m` を参照してください。

注 - `power` コマンドに `-sb` を指定しても、システムボードのハウスキーピング電力は制御されません。またオペレーティングシステムを実行中のドメインに属するシステムボードの電源を切断することもできません。

`-cb 0|1` 指定した制御ボード (背面の場合は 0、前面は 1) の電源を切断します。または、その電源装置への電力供給の状態を表示するか、電力供給を有効にします。アクティブな制御ボードが切断できるのは、除去の対象となる非アクティブの制御ボードだけです。`power` コマンドはこれら両方の電源は切断しません。2 番目の制御ボードの電源切断後、制御ボードの状態を取得するために `power` コマンドを使用しないでください。レジスタを読み込むことはできません。制御ボードを除去する前に、そのボードの LED を目視検査してください。

-csb *csb_list* 指定したセンタープレーンサポートボードの電源を投入する、切断する、あるいはその電源装置への電力供給を有効にします。*csb_list* (センタープレーンサポートボードのリスト) はスペースで区切られた整数のリストであり、0 (背面) または 1 (前面) です。

注 - **-csb** を指定しても、センタープレーンサポートボードのハウスキューピング電力は制御されません。

-p *p_list* リモートから制御される AC シーケンサおよびその接続周辺機器の電源を投入または切断するか、あるいはその電源への電力供給を有効にします。*p_list* はスペースで区切られた 0 ~ 4 の整数のリストであり、周辺機器用の配電装置を制御するホスト上のスイッチ (リモート電源制御ユニット) を示します。購入先の技術者の場合は、詳細は、『Sun Enterprise 10000 System Overview』(サービスドキュメントセットに含まれています) を参照してください。

-all すべてのシステムボード (センタープレーンサポートボードを含む) の電源を投入または切断します。**-sb** または **-csb** (あるいはこれらの両方) と一緒に **-all** を指定した場合は、**-all** が優先されます。

-d (Debug モード) 冗長なトレースメッセージを表示します。**-d** と **-q** を同時に指定することはできません。

-on 環境変数 **SUNW_HOSTNAME** で指定されたドメインに属するシステムボード (センタープレーンサポートボードを含む) の電源を投入します。**-sb**、**-csb**、**-all** も参照してください。

power -on は指定されたボードの電源を投入してから、システムに問い合わせ、システムに供給されている電力が十分であるかどうかを判別します。十分であれば、何もしません。十分でなければ、ボードの電源を切断して、メッセージを表示します。

- `-off` または
`-f -off` 環境変数 `SUNW_HOSTNAME` で指定されたドメインに属する個々のセンタープレーンサポートボードとシステムボードの電源を切断します。
`-sb`、`-csb`、`-cb`、`-p`、`-ps`、`-all` も参照してください。
`power -off` が指定されたボードの電源を切断できるのは、該当ドメインがオペレーティングシステムを実行していないときに限られます。ドメインがオペレーティングシステムを実行している場合は、`power -off` コマンドはそのことを知らせるメッセージを表示し、電源の切断は**行いません**。このため、ユーザーは正規の手順でドメインをシャットダウンしてから、電源を切断することができます。この保護機能を無効にし、ドメインの電源を即座に切断したい場合は、`-f -off` オプションを使用します。
- `-off -B` Sun Enterprise 10000 のホストキャビネットの電源およびリモートから制御している入出力キャビネットの電源を切断します。電源を再投入するには、48 ボルト電力を供給している AC 入力モジュール上のスイッチ (キャビネットあたり最大 4 個) を手動でオンにしなければなりません。
- `-ps ps_list` 指定した 48 ボルト電源を確認・切断します (購入先の技術者用)。またはその電源装置への電力供給を有効にします。`ps_list` は 48 ボルト電源を示す 1 つまたは複数の整数であり、有効範囲は `-0 ~ 7` です。- を付加すると、確認または無効にできます。

注 - すべての電源装置を切断する場合は、上記に示した `-off -B` オプションを使用してください。

Sun Enterprise 10000 のデュアル電源グリッドは最大 16 個の電源装置をサポートします。電源装置は 2 つのグリッドに分けられ、各グリッドは最大 8 個の電源装置で構成されます。主グリッド内の電源装置は、第 2 グリッド内の電源装置に関連付けられています。たとえば、電源装置 0 と 8、電源装置 1 と 9、電源装置 2 と 10 というように関連付けられます。`power` コマンドで 1 つの電源装置を切断した場合、指定した電源装置とその電源装置に関連付けられている電源装置の両方が切断されます。電源装置 0 ~ 7 を切断すれば、すべての電源装置を切断できます。

power -off -ps *ps_list* コマンドを実行すると、次のメッセージが表示されます。

```
WARNING: This system requires x 48V power
supplies for a current load of y system boards
(at N+1 redundancy). You are disabling z out of
w 48V power supplies. This system will be able
to normally operate a load of u system boards
(including N+1 redundancy). Continue? (y/n)
```

x は必要な電源装置の数です。

y はシステムに装着されているシステムボードの数です。

z は切断する電源装置の数です。

w はキャビネットに装着されている電源装置の数です。

u は指定されたシステムボードを切断した後にサポートされるシステムボードの数です。

注意 - 切断した 48 ボルト電源の数が多すぎると、システムがクラッシュすることがあります。警告メッセージの内容および下記のチャートを参照して、指定しようとする電源を無事切断できるかどうかを判断してください。

- 1 個のシステムボードは 4 個の電源を必要とする。
- 2 個のシステムボードは 4 個の電源を必要とする。
- 3 個のシステムボードは 5 個の電源を必要とする。
- 4 個のシステムボードは 5 個の電源を必要とする。
- 5 個のシステムボードは 5 個の電源を必要とする。
- 6 個のシステムボードは 6 個の電源を必要とする。
- 7 個のシステムボードは 6 個の電源を必要とする。
- 8 個のシステムボードは 6 個の電源を必要とする。
- 9 個のシステムボードは 7 個の電源を必要とする。
- 10 個のシステムボードは 7 個の電源を必要とする。
- 11 個のシステムボードは 7 個の電源を必要とする。
- 12 個のシステムボードは 7 個の電源を必要とする。
- 13 個のシステムボードは 8 個の電源を必要とする。
- 14 個のシステムボードは 8 個の電源を必要とする。
- 15 個のシステムボードは 8 個の電源を必要とする。
- 16 個のシステムボードは 8 個の電源を必要とする。

-q (Quiet モード) トラフィックの標準出力を抑止します。
-q と -d を同時に指定することはできません。

-v (Validate モード) 環境変数 SUNW_HOSTNAME で指定されたドメインに属するシステムボードとセンターブレイクサポートボードの個々の電源を検査し、有効になっているかどうかを判別します。
-v オプションはスクリプトで使用します。ターゲットデバイスの電源がすべて入っている場合は、終了コード 0 が生成されます。-v は電源制御ビットを検査するだけであり、電圧を検査するわけではありません。
-sb や -csb、あるいは -p と一緒に -v を使用すれば、検査の対象となるデバイスを指定できます。

-m ssp_resource(4) ファイルに保存されている現在の限界値を表示します。

```
-m margin_list -s -sb sb_list
```

購入先の技術者専用です。

```
-m margin_list -sb sb_list
```

購入先の技術者専用です。

power -m margin_list -s sb | csb コマンドは、すべてのシステムボードについて、*margin_list* に指定された値を、*ssp_resource(4)* ファイルに設定します。*-s* も指定されている場合は、変更が直ちに有効になります。指定されていない場合は、次回ボードの電源を投入したときに有効になります。*-s* を使用した場合には、*sb* の前にハイフンを挿入する必要がない点に注意してください。

注意 - 電源装置の限界値を $\pm 5\%$ 以上に設定すると、ハードウェアに損傷を与えることがあります。限界値をシステムボードの *Vcore* の値に設定することはできません。

power -s と **power -sb** では、*margin_list* を指定します。このリストは以下のいずれかの形式であり、複数の項目はスペースで区切ります。

```
supply.+margin
```

```
supply.margin (+margin と同じ)
```

```
supply.-margin
```

margin は 1 ~ 5 の整数であり、入力電圧値を基準として調整されるべき限界値を、パーセントで表します。

supply はキーワード *Vcc*、*Vdd* または *Vcore* で示され、システムボードの電圧を次のように表します。

```
Vcc, 5 ボルト
```

```
Vdd, 3.3 ボルト
```

このコマンドには 2 つの形式があり、それぞれ限界値の使用法が異なります。即時限界値は、**現行の電圧レベル**を、指定された限界値 (パーセント) によって調整します。限界値の自動設定の場合は、入力電圧レベルを、指定された限界値 (パーセント) によって調整します。

使用例

例 1 電源を投入する

システムボード 0 と 2 の電源を投入します。

```
power -on -sb 0 2
```

例 2 電源を切断する

環境変数 `SUNW_HOSTNAME` によって指定されたドメインに属するすべてのシステムボードの電源を切断します。

```
power -off
```

例 3 電圧の限界値を変更する

すべてのシステムボードについて、電圧の限界値を変更します。`Vcc` は 2% 増、`Vdd` は 2% 減となります。ただし、限界値の変更は直ちに有効になります。

```
power -m Vcc.+2 Vdd.-2 -s sb
```

例 4 電圧値を増加させる

システムボード 5、6、7 の `Vcc`、`Vdd` の電圧値をそれぞれ 3% 増やします。この変更は、システムボードに電源が再投入されたときに有効になります。

```
power -m Vcc.+3 Vdd.+3 -sb 5 6 7
```

例 5 周辺機器の電源を投入する

ホスト上のリモート電源制御ユニット (またはスイッチ) 2 と 3 によって、リモートから制御している周辺機器の電源を投入します。

```
power -p 2 3 -on
```

名前	redx - リモートエミュレーションデバッガ
形式	redx [-chnlq?][file [arg]...] redx [-q] -x redx_interpreter_cmd...
注意	ソフトウェアのインストールと更新の場合を除き、このコマンドは購入先の技術者のみが使用します。サンから提供されたインストールの説明書にこのコマンドの使用法が指示されている場合は、説明のとおりに行ってください。このコマンドの使用を誤ると、オペレーティングシステムに重大な障害を引き起こす場合があります。
機能説明	<p>redx は Sun Enterprise 10000 システムをデバッグし保守するためのサポートプログラムです。redx は POST プログラムと関係していますが、他の低レベルのハードウェアやファームウェアのデバッグにも使用されます。</p> <p>コマンドインタプリタ redx は、スクリプトから、あるいは -x オプションを指定した場合にはコマンド行から、コマンドを対話式に読み込みます。redx には詳細なオンラインヘルプが備わっています。プログラム内部からヘルプを呼び出すには、help または ? を入力してください。使用することができるコマンド名の一覧と、他のヘルプ機能に関する説明が表示されます。</p> <p>起動時に、redx はオプションのスタートアップスクリプト .redxrc を読み込みます。このファイルがカレントディレクトリに存在しない場合は、ユーザーのホームディレクトリを探します。どちらのディレクトリにもファイルが存在しない場合、あるいは -n オプションが指定されている場合は、スクリプトは使用されません。</p>
オプション	<p>ダッシュ付きの引数は、-qn のようにグループ化するか、あるいは -q -n のように個々に指定することができます。-x を除き、引数の順番は任意です。-x はダッシュ付き引数の最後に指定してください。数値引数は 10 進法で表記しますが、先頭に 0x または x を付けることによって 16 進法で表記することもできます。</p> <p>-c 入出力用の curses ライブラリ関数を使用しません。このモードでは、コマンドの履歴機能とスクロール可能な出力ウィンドウ機能は使用できません。redx を -c 付きで使用するのは、SSP へのシンプルな tty 接続が可能な場合に限ってください。</p> <p>-h ヘルプリストを出力します。</p>

-l	Local モードを指定します。このモードでは JTAG へのアクセスは行われません。このオプションは、関数を構文解析する場合や、安全のためにハードウェアのダンプファイルを オフラインで調べる場合に使用します。
-n	スタートアップファイル <code>.redxrc</code> を読み込みません。
-q	省略 (Quiet) モードを使用し、通常の画面出力を抑止します。
-x <i>redx_interpreter_cmd ...</i>	指定したインタプリタコマンドをコマンド行から実行し、そして終了します。シェル固有の文字はエスケープする必要があります (<code>sh(1)</code> を参照)。このオプションは <code>-n</code> の使用を前提とします。
-?	ヘルプリストを出力します。

使用例

例 1 オプションに関する説明を表示します。

```
redx -?
```

例 2 `-x` に関する詳細情報を表示します。

```
redx -x ?
```

名前	runcmdsync - フェイルオーバー後の回復のための指定されたスクリプトの準備										
形式	runcmdsync <i>script_name</i> [<i>parameters</i>]										
機能説明	runcmdsync(1M) コマンドは、フェイルオーバー後の自動同期 (回復) 向けに指定されたスクリプトを準備します。このコマンドは、回復するスクリプトを識別するためのコマンド同期記述子を生成します。この記述子は、フェイルオーバーの後に再起動するスクリプトを特定するコマンド同期リストに追加されます。また、 runcmdsync コマンドは、スクリプトが終了するとコマンド同期リストからこの記述子を削除します。										
オプション	以下のオプションをサポートしています。 <table border="0"> <tr> <td style="padding-right: 20px;"><i>script_name</i></td> <td>コマンドの同期をとるために作成するスクリプトを指定します。</td> </tr> <tr> <td><i>parameters</i></td> <td>指定されたスクリプトに関連するオプションとパラメータを指定します。これらのパラメータはスペア SSP に格納され、フェイルオーバーの後に指定されたコマンドとスクリプトを再起動するときに使用されます。</td> </tr> </table>	<i>script_name</i>	コマンドの同期をとるために作成するスクリプトを指定します。	<i>parameters</i>	指定されたスクリプトに関連するオプションとパラメータを指定します。これらのパラメータはスペア SSP に格納され、フェイルオーバーの後に指定されたコマンドとスクリプトを再起動するときに使用されます。						
<i>script_name</i>	コマンドの同期をとるために作成するスクリプトを指定します。										
<i>parameters</i>	指定されたスクリプトに関連するオプションとパラメータを指定します。これらのパラメータはスペア SSP に格納され、フェイルオーバーの後に指定されたコマンドとスクリプトを再起動するときに使用されます。										
終了ステータス	以下の終了ステータスが返されます。 <table border="0"> <tr> <td style="padding-right: 20px;">0</td> <td>正常に終了しました。</td> </tr> <tr> <td>0 以外</td> <td>エラーが発生しました。</td> </tr> </table>	0	正常に終了しました。	0 以外	エラーが発生しました。						
0	正常に終了しました。										
0 以外	エラーが発生しました。										
属性	以下の属性の記述子については、 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>Architecture</td> <td>すべて</td> </tr> <tr> <td>Availability</td> <td>SUNWuessp</td> </tr> <tr> <td>Stability</td> <td>Evolving</td> </tr> <tr> <td>MT-Level</td> <td>Safe</td> </tr> </tbody> </table>	属性タイプ	属性値	Architecture	すべて	Availability	SUNWuessp	Stability	Evolving	MT-Level	Safe
属性タイプ	属性値										
Architecture	すべて										
Availability	SUNWuessp										
Stability	Evolving										
MT-Level	Safe										
関連項目	cancelcmdsync(1M) 、 initcmdsync(1M) 、 savecmdsync(1M) 、 showcmdsync(1M)										

名前	savecmdsnc, cmdsync, cancelcmdsnc, initcmdsnc - コマンド同期コマンド
形式	<pre>cancelcmdsnc cmdsync_descriptor initcmdsnc script_name [parameters] savecmdsnc -M identifier cmdsync_descriptor</pre>
機能説明	<p>一連のコマンド同期コマンドは、連動して SSP フェイルオーバーによって割り込まれたユーザー定義のスキプトの回復を制御します。コマンドの同期を有効にするには、ユーザー定義のスキプトに以下のコマンドを挿入します。</p> <ul style="list-style-type: none"> ■ <code>initcmdsnc</code> は、回復するスキプトを識別するためのコマンド同期記述子を生成します。 <p>この記述子は、コマンド同期リストに追加されます。この記述子によって、フェイルオーバーの後に新しいメイン SSP で再起動するスキプトとコマンドを識別します。</p> <ul style="list-style-type: none"> ■ <code>savecmdsnc</code> は、フェイルオーバーの後に処理を再開できるスキプト上の場所を特定するマーカーを追加します。 ■ <code>cancelcmdsnc</code> は、コマンド同期リストからコマンド同期記述子を削除します。削除することで、確実にスキプトは一度だけ実行され、次のフェイルオーバーの後には実行されません。 <p>スキプトのすべての終了パスに <code>cancelcmdsnc</code> シーケンスを記述して、コマンド同期リストから記述子を確実に削除します。記述子を削除しない状態でフェイルオーバーが起動すると、スキプトは新しいメイン SSP で再実行されます。</p> <hr/> <p>注 - コマンドの同期を有効にするには、<code>initcmdsnc</code> と <code>cancelcmdsnc</code> の両方のシーケンスをスキプトに記述する必要があります。<code>savecmdsnc</code> コマンドは省略可能ですが、処理を再開できるスキプト上の特定の場所を指定する場合のみ使用します。</p> <hr/>
オプション	<p>以下のオプションをサポートしています。</p> <p><code>cmdsync_descriptor</code> ユーザー定義のスキプトを識別するコマンド同期記述子を指定します。この記述子は、<code>initcmdsnc</code> コマンドによって返される標準出力値です。</p> <p><code>-M identifier</code> フェイルオーバーの後に再開できるスキプト上の場所を指定します。この識別子には、必ず正の整数を使用します。</p>

拡張機能説明

parameters ユーザー定義のスクリプトに関連するオプションとパラメタを指定します。これらのパラメタはスペア SSP に格納され、フェイルオーバーの後に指定されたスクリプトを再起動するときに使用されます。

script_name 同期させるユーザー定義のスクリプトの名前を指定します。

コマンド同期コマンドは、ユーザー定義スクリプトにおける決められた論理上の位置に挿入します。

たとえば Korn シェルスクリプトの場合は、以下のような構造になります。

```
# MAIN CODE STARTS HERE
# Be sure to use a cleanup procedure to handle any interrupts.
# Use the cancelcmdsnc to remove the script from the command
# synchronization list. Otherwise, the command will get restarted
# on the new main SSP.
#
clean_up () {
    cancelcmdsnc $desc
    exit
}
# Declare the clean_up function to capture system signals
# and cleanup.
trap "clean_up" INT HUP TERM QUIT PWR URG
goto_label=1
# Process the arguments, capturing the -M marker point if provided
#
for arg in $*; do
    case $arg in
        -M )
            goto_label=$arg;;
        .
        .
    esac
done
# Place this script and all its parameters in the command synchronization
# list, which indicates the commands to be restarted after an SSP failover.
#
# NOTE: The script must be executable by user ssp and reside in the same
# directory on both the main and the spare SSP. If the command is not
# part of the defined PATH for user ssp, the absolute filename must be
# passed with the initcmdsnc command.
#
initcmdsnc script_name parameters
# The marker point is stored in the goto_label variable.
# Keep executing this script until all cases have been processed or an
# error is detected.
#
while (( $goto_label != 0 )) ; do
    #
    # Each case should represent a synchronization point in the script.
    #
    case $goto_label in
        #
        # Step 1: Do something
        #
        1 )
            do_something
            .
            .
            # Execute the savecmdsnc command with the script's
            # descriptor and a unique marker to save the position.
            # If a failover occurs here, the commands
            # represented in the next goto_label (2) will be
```

```

        # resumed.
        #
        savecmdsnc -M $(( $goto_label + 1 )) $desc
        goto_label=$(( $goto_label + 1 ))
        ;;
    #
    # Step 2: Do more things
    #
    2 )
        do_more_things
        .
        .
        savecmdsnc -M $(( $goto_label + 1 )) $desc
        goto_label=$(( $goto_label + 1 ))
        ;;
    #
    # Step 3: Finish the last step and set the goto_label to 0
    # so that the script ends.
    3 )
        finish_last_step
        .
        .
        goto_label=0
        ;;
    esac
done
# END OF MAIN CODE
# Remember to execute cancelcmdsnc to remove the script from the
# command synchronization list. Otherwise, the command will be restarted
# after the failover.
#
cancelcmdsnc $desc

```

終了ステータス

以下の終了ステータスが返されます。

0	正常に終了しました。
0 以外	エラーが発生しました。

注 - `initcmdsnc` に対する標準出力は、コマンド同期記述子で構成されます。また、フェイルオーバーが起動しない場合 (フェイルオーバーの後、またはシングル SSP 環境の場合) は、同期コマンドを含むスクリプトによってプラットフォームログファイルにエラーメッセージが生成され、0 以外の終了コードが返されます。このエラーメッセージは、無視しても問題はありません。

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

関連項目

runcmdsnc(1M), showcndsnc(1M)

ディレクトリ /opt/SUNWssp/examles/cmdsnc の (同期コマンドを含む) ユーザー定義スクリプトの例

名前	setdatasync - データの同期で使用するデータ伝播リストの変更
形式	<pre>setdatasync [-i interval] schedule filename</pre> <pre>setdatasync cancel filename</pre> <pre>setdatasync push filename</pre> <pre>setdatasync backup</pre> <pre>setdatasync clean</pre>
機能説明	<p>setdatasync を使用すると、ユーザーが作成したファイルを指定してデータ伝播リスト (/var/opt/SUNWssp/.ssp_private/user_file_list) へ追加したり、このリストから削除することができます。自動フェイルオーバーのためにデータ同期プロセスの一環としてメイン SSP からスペア SSP へコピーするファイルは、このリストで指定します。指定されたユーザーファイルとファイルの保存先のディレクトリは、ユーザー ssp に対する読み取り権と書き込み権が必要です。</p> <hr/> <p>注 - データの同期は、/tmp ディレクトリ上の使用可能なディスク容量を利用して、メイン SSP からスペア SSP へファイルをコピーします。コピーするファイルの容量が /tmp ディレクトリよりも大きい場合は、このファイルを伝播することはできません。データ同期バックアップファイル (ds_backup.cpio) が /tmp 上の使用可能な容量よりも大きくなる場合は、データの伝播を実行する前にバックアップファイルのサイズを小さくする必要があります。ssp_backup.cpio ファイルのサイズはデータ同期バックアップファイルのサイズを示します。以下のファイルは削除しても問題はありません。</p> <ul style="list-style-type: none"> ■ SSPLOGGER/messages.x ■ SSPLOGGER/domain/Edd-recovery files ■ SSPLOGGER/domain/messages.x ■ SSPLOGGER/domain/netcon.x ■ SSPLOGGER/post/files <p>x はファイルのアーカイブ番号です。これらのファイルはフェイルオーバーの後で新しいメイン SSP からスペア SSP へ伝播されるため、ファイルが更新されないようにメイン SSP とスペア SSP の両方から削除する必要があります。</p> <hr/> <p>データ同期プロセスは、メイン SSP のユーザーが作成したファイルに対する変更を確認します。最後に伝播された後でメイン SSP のユーザーが作成したファイルが変更されると、これらのファイルはスペア SSP へ再伝播されます。デフォルトでは、指定されたファイルを 60 分ごとに確認しますが、setdatasync を使用してユーザーファイルの変更を確認する間隔を指定することができます。</p>

注 - ファイルがメイン SSP からスペア SSP へ伝播されると、このファイルはメイン SSP が更新された場合のみスペア SSP へ再伝播されます。伝播されたファイルをスペア SSP から削除した場合は、メイン SSP の対応するファイルが変更されるまでは、そのファイルは自動的に伝播されません。

setdatasync を使用すると、以下の処理も実行できます。

- 指定したファイルをデータ伝播リストへ追加せずにスペア SSP へ伝播する。
- メイン SSP とスペア SSP の SSP 設定ファイルの同期をとり直す。
- データ伝播リスト上のエントリと一時ファイルをすべて削除し、データ伝播リストを削除する (SSP が 1 台の場合に有効)。

オプション

以下のオプションをサポートしています。

backup メイン SSP のバックアップをとり、バックアップデータをメイン SSP からスペア SSP へ移動して、スペア SSP 上でバックアップデータを復元します。

このコマンドによるバックアップは、/tftpboot ディレクトリをバックアップしない点を除けば、ssp_backup を実行して作成されるバックアップと同じです。データ同期バックアップでは、以下のファイルを復元しません。

- /var/opt/SUNWssp/.ssp_private/machine_server_fifo
- /var/opt/SUNWssp/adm/messages
- /var/opt/SUNWssp/adm/messages.dsbk
- /var/opt/SUNWssp/.ssp_private/user_file_list
- /var/opt/SUNWssp/.ssp_private/.ds_queue

作成されるバックアップファイルの容量が /tmp ディレクトリに割り当てられているディスク容量を越えると、データ同期バックアップは失敗します。バックアップファイルの容量を小さくする方法については、「機能説明」の「注」を参照してください。

<code>cancel filename</code>	指定したファイルをデータ伝播リストから削除する(取消す)ことで、このファイルをスペア SSP へ伝播されないようにします。このオプションは、指定したファイルを実際にスペア SSP から削除するものではありません。ファイル名は絶対パスで指定してください。また、別のファイルに対するシンボリックリンクファイルにすることはできません。
<code>clean</code>	データ伝播リスト上のエントリと一時ファイルをすべて削除して、リストを削除します。このオプションは、シングル SSP 構成でディスク容量を管理しており、データ伝播リストの量がかなり増えて必要以上にディスク容量を使用する可能性がある場合に役立ちます。 このオプションを使用できるのは、1つの SSP で構成されている場合に限りです。定期的に <code>setdatasync clean</code> コマンドを実行して、 <code>/tmp</code> ディレクトリに蓄積される SSP メッセージとログのファイル数を減らしてください。 デュアル SSP で構成されている場合は、このオプションを使用しないでください。使用すると、メイン SSP とスペア SSP の間でデータが一致しなくなる可能性があります。
<code>-i interval</code>	指定されたファイルに対する変更を確認する間隔を指定します。デフォルトは 60 分です。間隔は 1 分から 1440 分 (24 時間) の範囲で設定することができます。
<code>push filename</code>	指定したファイルをデータ伝播リストへ追加せずにスペア SSP へ伝播(プッシュ)します。ファイル名は絶対パスで指定してください。また、別のファイルに対するシンボリックリンクファイルにすることはできません。
<code>schedule filename</code>	指定したファイルをデータ伝播リストへ追加します。ファイル名は絶対パスで指定してください。また、別のファイルに対するシンボリックリンクファイルにすることはできません。データ同期プロセスの間にファイルはスペア SSP 上の同じ絶対パスへ伝播されます。

終了ステータス

以下の終了ステータスが返されます。

0	正常に終了しました。
0 以外	エラーが発生しました。

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

ファイル

以下のファイルをサポートしています。

/var/opt/SUNWssp/.ssp_private/user_file_list
データ伝播リスト

関連項目

datasyncd(1M)、showcmdsnc(1M)

Sun Enterprise 10000 SSP ユーザーマニュアル

名前	setfailover - SSP または制御ボードに対する自動フェイルオーバー状態の変更
形式	setfailover [-t cb] on off force setfailover -m [<i>memory_threshold</i>] setfailover -d [<i>disk_space_threshold</i>]
注意	フェイルオーバーを強制起動する前に、必ずメイン SSP とスペア SSP を同期させてください。両方の SSP を同期させるには、 setdatasync(1M) コマンドを使用します。
機能説明	<p>setfailover を使用すると、自動フェイルオーバーの対象を SSP と制御ボードのいずれかに設定し、自動フェイルオーバーを使用可能または使用不可にしたり、強制起動することができます。デフォルトでは、自動フェイルオーバーの対象は SSP に設定されています。</p> <p>fod デーモンが SSP フェイルオーバーを起動するかどうかを判断する際に使用する仮想記憶とディスク容量のしきい値パラメタの照会や変更にも setfailover を使用できます。</p> <p>setfailover は必ずメイン SSP 上で実行してください。</p>

オプション

以下のオプションをサポートしています。

force

-t cb が指定されている場合に、利用可能なスペア SSP または制御ボードに対する自動フェイルオーバーを強制起動します。

-t cb が指定されている場合は、JTAG インタフェースがスペア制御ボードへ移動しますが、システムクロックは移動しません。システムクロックは、有効なドメインがない場合に移動します。

強制フェイルオーバーを行う前に setfailover コマンドを使用すると、データの同期状態を調べることができます。以下のいずれかの状態となっていると、強制フェイルオーバーは行われません。

- データ同期のバックアップを行っている (アクティブアーカイブと呼びます) 場合
- メイン SSP からスペア SSP へファイルを伝播中 (アクティブファイルと呼びます) の場合
- データ同期キュー内に少なくとも 1 つ以上のファイルが入っている場合

showdatasync(1M) コマンドを実行すると、データ同期の状態情報 (データ同期のバックアップ操作やファイルの伝播操作が行われているかどうか、ファイルが同期キューに入っているかどうか) を調べることができます。データ同期の詳細については、showdatasync(1M) に関するマニュアルページおよび『Sun Enterprise 10000 SSP 3.5 ユーザーマニュアル』を参照してください。

off

-t cb が指定されている場合に、SSP または制御ボードに対する自動フェイルオーバーを使用不可にします。on に設定して使用可能にするまでは、自動フェイルオーバーは使用不可のままです。

on

オペレータの要求またはシステムの障害によってフェイルオーバーが使用不可となったシステムの自動フェイルオーバーを使用可能にします。

setfailover はフェイルオーバーを使用可能にする前に接続状態を確認します。接続に障害が発生している場合は、フェイルオーバーを使用可能にしません。

-t *cb* 自動フェイルオーバーの対象を制御ボードに設定します。

-m [*memory_threshold*] *ssp_resource* ファイルに仮想記憶のしきい値を設定します。この場合の *memory_threshold* は更新値を K バイトで現した値です。-m のみを指定すると、*ssp_resource* ファイルに保存されている現在の仮想記憶のしきい値が表示されます。

-d [*disk_space_threshold*] *ssp_resource* ファイルにディスク容量のしきい値を設定します。この場合の *disk_space_threshold* は更新値を K バイトで現した値です。-d のみを指定すると、*ssp_resource* ファイルに保存されている現在のディスク容量のしきい値が表示されます。

終了ステータス

以下の終了ステータスが返されます。

0 正常に終了しました。
0 以外 エラーが発生しました。

属性

以下の属性の記述子については、`attributes(5)` を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

ファイル

`$SSPVAR/.ssp_private/ssp_resource` SSP リソースファイル

関連項目

`fod(1M)`、`setdatasync(1M)`、`showdatasync(1M)`、`showfailover(1M)`、`ssp_resource(4)`

『Sun Enterprise 10000 SSP ユーザーマニュアル』

名前	showcmdsnc - 現在のコマンド同期リストを表示										
形式	showcmdsnc										
機能説明	<p>自動 SSP フェイルオーバー後に再起動する必要があるコマンドやスクリプトを特定するために、showcmdsnc を使用してスペア SSP で使用されるコマンド同期リストを表示します。</p> <p>コマンド同期リストは <i>Descriptor</i>、<i>Identifier</i>、<i>Cmd</i> の形式で表示されます。それぞれの意味は以下のとおりです。</p> <p><i>Descriptor</i> 特定のスクリプトを表すコマンド同期記述子を示します。</p> <p><i>Identifier</i> 自動フェイルオーバーの起動後に新しいメイン SSP でスクリプトを再開できるスクリプト上のマーカー位置を示します。-1 はスクリプト上にマーカー位置が指定されていないことを示します。</p> <p><i>Cmd</i> 再起動するスクリプトの名前を示します。</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 正常に終了しました。</p> <p>0 以外 エラーが発生しました。</p>										
属性	<p>以下の属性の記述子については、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>Architecture</td> <td>すべて</td> </tr> <tr> <td>Availability</td> <td>SUNWuessp</td> </tr> <tr> <td>Stability</td> <td>Evolving</td> </tr> <tr> <td>MT-Level</td> <td>Safe</td> </tr> </tbody> </table>	属性タイプ	属性値	Architecture	すべて	Availability	SUNWuessp	Stability	Evolving	MT-Level	Safe
属性タイプ	属性値										
Architecture	すべて										
Availability	SUNWuessp										
Stability	Evolving										
MT-Level	Safe										
使用例	<p>例 1 コマンド同期リストの例</p> <pre>% showcmdsnc DESCRIPTOR IDENTIFIER CMD 0 -1 c1 a1 a2</pre>										

showcmdsycn(1M)

保守コマンド

関連項目

cancelcmdsycn(1M)、initcmdsycn(1M)、runccmdsycn(1M)、
savecmdsycn(1M)

名前	showdatasync - フェイルオーバーの起動前に SSP のデータ同期状態を表示
形式	showdatasync [-l -Q]
機能説明	showdatacync を使用すると、メイン SSP からスペア SSP へ伝播 (コピー) されたファイルの現在の状態がわかります。データの伝播によってスペア SSP のデータはメイン SSP のデータとの同期がとられるため、自動フェイルオーバーが起動すると、スペア SSP はメイン SSP の最新の状態に合わせられます。
オプション	<p>以下のオプションをサポートしています。</p> <p>-l 現在のデータ伝播リストにあるファイルを一覧表示します。表示される内容の詳細については、「拡張機能説明」を参照してください。</p> <p>-Q 伝播待ちファイルを一覧表示します。各ファイル名には、ファイルの絶対パスが表示されます。</p>
拡張機能説明	<p>ここでは showdatacync コマンドで表示される内容を説明します。</p> <p>showdatacync コマンドでオプションを指定しない場合は、以下の内容が表示されます。</p> <pre>File Propagation Status: Active File: Queued files:</pre> <p>それぞれの意味は以下のとおりです。</p> <p>File Propagation Status 現在のデータ同期状態を表示します。</p> <ul style="list-style-type: none"> ■ Active は、データ同期プロセスが有効で、正常に機能していることを示します。 ■ Active Archive は、データ同期のバックアップを行っていることを示します。 ■ Disabled は、自動 SSP フェイルオーバーが使用不可になっているため、データ同期プロセスが無効であることを示します。 ■ Failed は、過去に SSP の障害が検出された場合でも、現在はデータ同期プロセスによってファイルをスペア SSP へ伝播できないことを示します。

Active File 現在伝播されているファイルの絶対パスか、接続がアイドル状態であることを示す - (ダッシュ) を表示します。

Queued files 伝播の対象となっていて、まだ処理されていないファイルの数を示します。

showdatasync コマンドで -1 を指定すると、データ伝播リスト上の各エントリが *Time Propagated*、*Interval*、*File* の形式で表示されます。それぞれの意味は以下のとおりです。

Time Propagated 最後にファイルがメイン SSP からスベア SSP へ伝播された時刻を示します。

Interval ユーザー定義ファイルへの変更を確認する間隔を分単位で示します。デフォルトの間隔は 60 分です。

File 伝播されるファイルの絶対パスと名前を示します。

終了ステータス

以下の終了ステータスが返されます。

0 正常に終了しました。
0 以外 エラーが発生しました。

属性

以下の属性の記述子については、attributes(5) を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

使用例

例 1 データ同期状態

```
% showdatasync
File Propagation State: ACTIVE
Active File:             -
Queued files:           0
```

例 2 データ同期バックアップ (/tmp/ds_backup.cpio) の状態

```
% showdatasync
File Propagation State: ACTIVE ARCHIVE
Active File:             -
Queued files:           0
```

例 3 データ同期リスト

```
% showdatasync -l
TIME PROPAGATED          INTERVAL      FILE
Mar 23 16:00:00          60          /tmp/t1
```

例 4 データ同期キュー

```
% showdatasync -Q
FILE
/tmp/t1
```

関連項目

setdatasync(1M)

『Sun Enterprise 10000 SSP ユーザーマニュアル』

-p bydevice|byboard|query|force

以下の指定に合わせてレポートを表示します。

bydevice— デバイスのタイプ (CPU、メモリー、入出力) ごとに出力をリストします。

byboard — システムボードごとに出力をリストします。この表示は、デバイスのタイプ (CPU、メモリー、入出力) ごとにまとめられた表形式となります。

query — システムボードを削除した結果を予測します。

force — オフラインによる照会を強制します。リソースを消費するアプリケーションやサブシステムに対して強制セマンティックの適用を要求して、システムリソースを放棄できるかどうかを予測します。[cfgadm\(1M\)](#) を参照してください。

-v

入出力デバイスをすべて表示します。管理対象の入出力デバイスだけでなく、管理対象以外の入出力デバイスも表示します。管理対象のデバイスについては、現在、管理しているリソースが表示されます。管理対象以外のデバイスは物理的に構成されていますが、これらが管理しているリソースは表示されません。また、管理対象以外のデバイスについては、使用状況に関する情報は表示されません。

オペランド

以下のオペランドをサポートしています。

board_id

ボード識別子。有効なボード識別子の範囲は SB0 ~ SB15 となります。

拡張機能説明

showdevices のフィールドは以下のとおりです。

domain	— ドメイン名
board	— ボード識別子
CPU:	
id	— プロセッサの ID
state	— プロセッサの状態
speed	— CPU の周波数 (MHz 単位)
ecache	— CPU の外部キャッシュサイズ (MB 単位)
query	— リソースについてのオフライン照会の結果
メモリー:	
board mem	— ボードに実装されているメモリーのサイズ (MB 単位)
perm mem	— ボードに実装されている再配置不可能なメモリーのサイズ (MB 単位)
base address	— ボードに実装されているメモリーの物理アドレス
domain mem	— システムメモリーのサイズ (MB 単位)
board	— ボード識別子
query	— リソースについてのオフライン照会の結果
メモリーのドレイン操作を行っている場合は、以下の情報が表示されません。	
target board	— ターゲットボードの識別子
deleted	— 削除済みのメモリーのサイズ (MB 単位)
remaining	— 削除するメモリーのサイズ (MB 単位)

入出力デバイス

device	— 入出力デバイスのインスタンス名
resource	— 管理対象のリソース名
usage	— リソースを使用しているインスタンスの説明
query	— リソースについてのオフライン照会の結果

使用例

例 1 システムボード SB5 に対する showdevices 実行例

```

ssp% showdevices -v SB5
CPU
----
domain  board  id    state  speed  ecache  usage
xf8-b4  SB5    20    online  333    4
xf8-b4  SB5    21    online  333    4
xf8-b4  SB5    22    online  333    4
xf8-b4  SB5    23    online  333    4
Memory
-----
domain  board  board  perm  base      domain
xf8-b4  SB5    memMB  memMB address  memMB
                2048   0      0x500000 6144

IO Devices
-----
domain  board  device  resource  usage
xf8-b4  SB5    qfe52
xf8-b4  SB5    qfe53
xf8-b4  SB5    qfe54
xf8-b4  SB5    qfe55
xf8-b4  SB5    sd195
xf8-b4  SB5    sd196

```

例 2 ドメイン xf8-b4 に対する showdevices 実行例

```

ssp% showdevices -v -d xf8-b4
CPU
---
domain  board  id    state  speed  ecache  usage
xf8-b4  SB4    16    online  333    4
xf8-b4  SB4    17    online  333    4
xf8-b4  SB4    18    online  333    4
xf8-b4  SB4    19    online  333    4
xf8-b4  SB5    20    online  333    4
xf8-b4  SB5    21    online  333    4
xf8-b4  SB5    22    online  333    4
xf8-b4  SB5    23    online  333    4
xf8-b4  SB6    24    online  333    4
xf8-b4  SB6    25    online  333    4
xf8-b4  SB6    26    online  333    4
xf8-b4  SB6    27    online  333    4
Memory
-----
domain  board  board  perm  base  domain
        board memMB memMB address memMB
xf8-b4  SB4    2048  648  0x600000  6144
xf8-b4  SB5    2048  0    0x500000  6144
xf8-b4  SB6    2048  0    0x400000  6144

IO Devices
-----
domain  board  device  resource  usage
xf8-b4  SB4    qfe0
xf8-b4  SB4    qfe1
xf8-b4  SB4    qfe2
xf8-b4  SB4    qfe3
xf8-b4  SB4    sd0     /dev/dsk/c0t0d0s0  mounted filesystem "/"
xf8-b4  SB4    sd0     /dev/dsk/c0t0d0s1  swap area
xf8-b4  SB4    sd0     /dev/dsk/c0t0d0s1  dump device (swap)
xf8-b4  SB4    sd1
xf8-b4  SB5    qfe52
xf8-b4  SB5    qfe53
xf8-b4  SB5    qfe54
xf8-b4  SB5    qfe55
xf8-b4  SB5    sd195
xf8-b4  SB5    sd196
xf8-b4  SB6    qfe16
xf8-b4  SB6    qfe17
xf8-b4  SB6    qfe18
xf8-b4  SB6    qfe19
xf8-b4  SB6    sd60
xf8-b4  SB6    sd61

```

例 3 システムボード SB5 に対するオフライン照会の結果表示例

```

ssp% showdevices -v SB5
CPU
---
domain  board  id    state  speed  ecache  query  usage:reason
xf8-b4  SB5    20    online 333    4
xf8-b4  SB5    21    online 333    4
xf8-b4  SB5    22    online 333    4
xf8-b4  SB5    23    online 333    4
Memory
-----
          board  perm  base      domain
domain  board  memMB memMB  address  memMB  query  usage:reason
xf8-b4  SB5    2048  0       0x500000 6144

IO Devices
-----
domain  board  device  resource          query  usage:reason
xf8-b4  SB5    qfe52
xf8-b4  SB5    qfe53
xf8-b4  SB5    qfe54
xf8-b4  SB5    qfe55
xf8-b4  SB5    sd195
xf8-b4  SB5    sd196

```

query フィールドには、リソースを削除したときの予測結果が表示されます(上記の例は表示していません)。

終了ステータス

以下の終了ステータスが返されます。

- | | |
|---|---------------------------------|
| 0 | 正常に終了しました。 |
| 1 | 指定されたドメインは無効です。 |
| 2 | 無効なオプションなど、コマンド行の入力エラーが検出されました。 |
| 3 | 複数のドメインが指定されました。 |
| 5 | ドメインとの通信中にエラーが発生しました。 |
| 6 | デバイス情報の処理中にエラーが発生しました。 |
| 7 | 記憶域割り当ての不具合などの内部エラーが発生しました。 |

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWssp
Stability	Evolving
MT - Level	Safe

関連項目

『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』
の addboard(1M)、deleteboard(1M)、moveboard(1M)、
rcfgadm(1M)

名前	showfailover - SSP と制御ボードのフェイルオーバー状態を表示
形式	showfailover [-r]
機能説明	showfailover は SSP と制御ボードに関する現在のフェイルオーバー状態を表示します。表示される内容の詳細については、「拡張機能説明」を参照してください。
オプション	<p>以下のオプションをサポートしています。</p> <p>-r SSP の役割を MAIN、SPARE、UNKOWN のいずれかに特定します。SSP が制御ボードと通信できないときには、SSP の起動に少し時間がかかります。この場合は SSP の役割を判断できないため、UNKOWN とみなされます。</p>
拡張機能説明	<p>showfailover コマンドでオプションを指定しない場合は、以下の内容が表示されます。</p> <ul style="list-style-type: none"> ■ フェイルオーバー状態 - SSP と制御ボードのフェイルオーバーの状態。以下のいずれかが表示されます。 <ul style="list-style-type: none"> ■ Active - 自動フェイルオーバーが使用可能で、正常に機能していることを示します。 ■ Disabled - オペレータの要求、またはフェイルオーバーの起動を妨げる障害により、自動フェイルオーバーが使用不可になっていることを示します。 ■ Failed - フェイルオーバーが起動したことを示します。フェイルオーバー後は、setfailover(1M) コマンドで再びフェイルオーバーを使用可能にするまでは「障害が発生した」状態とみなされません。すべての接続を修正し、Failover connection map (以下に説明) で GOOD と判定されている場合でも、手動で使用可能に設定し直す必要があります。 ■ Failover connection map - フェイルオーバーデーモン (fod) によって監視される制御ボードの接続リンクの状態。接続リンクの状態は、接続が正常に機能していることを示す GOOD と、接続が機能していないことを示す FAILED のいずれかが表示されます。 ■ SSP/CB Host Information - SSP、制御ボード、JTAG インタフェースとシステムクロックを管理する制御ボードの名前。
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 正常に終了しました。</p> <p>0 以外 エラーが発生しました。</p>

属性

以下の属性の記述子については、attributes(5)を参照してください。

属性タイプ	属性値
Architecture	すべて
Availability	SUNWuessp
Stability	Evolving
MT-Level	Safe

使用例

例 1 フェイルオーバーの状態

以下の例では、スペア SSP に障害が発生し、SSP フェイルオーバーが使用不可になっています。

```

ssp% showfailover
Failover State:
  SSP Failover:  Disabled
  CB Failover:   Active
Failover Connection Map:
  Main SSP to Spare SSP thru Main Hub:      FAILED
  Main SSP to Spare SSP thru Spare Hub:     FAILED
  Main SSP to Primary Control Board:        GOOD
  Main SSP to Spare Control Board:          GOOD
  Spare SSP to Main SSP thru Main Hub:      FAILED
  Spare SSP to Main SSP thru Spare Hub:     FAILED
  Spare SSP to Primary Control Board:       FAILED
  Spare SSP to Spare Control Board:         FAILED
SSP/CB Host Information
  Main SSP:                                  xf12-ssp
  Spare SSP:                                 xf12-ssp2
  Primary Control Board (JTAG source):      xf12-cb1
  Spare Control Board:                      xf12-cb0
  System Clock source:                     xf12-cb1

```

例 2 SSP の役割

```

ssp% showfailover -r
SPARE

```

関連項目

setfailover(1M)

名前	showusage - DR およびボードリソース情報の表示
形式	showusage -b <i>board_number</i> -r <i>report_type</i> [-r <i>report_type</i> ...] showusage -b <i>board_number</i> -s
機能説明	<p>showusage に -r オプションを指定して実行すると、標準出力にボードリソース、システムリソース、および DR についての情報が出力されます。情報は表形式で出力されます。</p> <p>showusage に -s オプションを指定して実行すると、指定したボードの状態を示す整数の終了ステータスが返されます。指定したボードがドメイン内にある場合は、showusage は標準出力にドメイン名を出力します。-s オプションは ADR スクリプトで使用します。また、ADR 操作の実行前に、ボードが ADR 操作に適切かどうかを判定するためにも使用できます。</p> <hr/> <p>注 - このコマンドを適用できるのは、DR モデル 2.0 を実行している Sun Enterprise 10000 のドメインに対してのみとなります。ドメインで実行されている DR モデルを調べるには、domain_status -m コマンドを使用します。</p> <hr/>

オプション

以下のオプションをサポートしています。

- b *board_number* 状態情報を取得するシステムボードの番号 (0 ~ 15)。
- r *report_type* 情報の表示形式。 *report_type* は、以下のいずれかです。
 CPU - ボードのプロセッサ情報を表示します。
 DRAIN - アクティブなドレイン操作の進行状況を表示します。
 IO - このボードに接続されているデバイスの情報を表示します。
 OBP - OBP が認識したボード構成を表示します。初期接続は行われたが、接続が完了していないボードで OBP 表示を使用できます。OBP 表示は、実行中のボードの CPU、MEM、および IO 表示ほど正確ではないことがあります。
 MEM - このボードのメモリー構成を表示します。
 UNSAFE - 指定したドメインで使用される、すべての保存停止された安全ではないデバイスを表示します。このオプションを使用すると、-b オプションは無視されます。
 複数のレポート形式を指定する場合は、-r オプションを複数指定する必要があります。たとえば、ボード 12 のプロセッサおよびデバイス情報を表示するには、次のコマンドを使用します。
showusage -b 12 -r CPU -r IO
- s ボードの状態を終了ステータスとして報告し、標準出力にドメイン名を出力します。指定したボードがドメインに属していない場合は、標準出力に何も出力されません。

終了ステータス

showusage コマンドは、指定したコマンドオプションに応じて、異なる終了ステータスを返します。

-s オプションを指定した場合は、次の終了値が返されます。

- 0 ボードが存在せず、ソフトウェアドメインにも含まれていません。ソフトウェアドメインとは、`domain_config` ファイルに表示されているアクティブドメインです。ただし、CPU は実行されていません。
- 1 ボードは存在しませんが、ソフトウェアドメインには含まれています。
- 2 ボードは存在しますが、電源が切断されており、ソフトウェアドメインに含まれていません。

- 3 ボードが存在し電源が投入されていますが、ソフトウェアドメインには含まれていません (ボードは接続可能です)。
- 4 ボードは存在しますが、電源が切断されています。ソフトウェアドメインには含まれています。
- 5 ボードが未設定 (ボードは存在し、電源が投入されている。ソフトウェアドメインには含まれているが、ハードウェアドメインには含まれていない) です。ハードウェアドメインはブートドメインです。
- 6 ボードは設定済み (ボードは存在し、電源が投入されている。ソフトウェアドメインとハードウェアドメインの両方に含まれている) ですが、DR 切り離し可能状態にありません。ボードの切り離しを妨げる条件については、『Sun Enterprise 10000 Dynamic Reconfiguration ユーザーマニュアル』を参照してください。
- 7 ボードは設定済み (ボードは存在し、電源が投入されている。ソフトウェアドメインとハードウェアドメインの両方に含まれている) で、切り離しがほぼ可能な状態にあります。ボードの切り離しを妨げる条件については、『Sun Enterprise 10000 Dynamic Reconfiguration ユーザーマニュアル』を参照してください。
- 8 ボードは DR モデル 3.0 のドメインに属しています。showusage コマンドはこのドメインに対しては使用できません。
- 15 DR ライブラリに動的にリンクできませんでした。
- 255 showusage 構文エラー
- r オプションを指定すると、次の終了値が返されます。
- 0 レポートが正常に表示されました。
- 255 showusage 構文エラー

エラーメッセージ

showusage が失敗した場合は、診断メッセージが標準エラーに出力されます。

関連項目

『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』の addboard(1M)、moveboard(1M)、deleteboard(1M)、dr(1M)、drshow(1M)

『Sun Enterprise 10000 Dynamic Reconfiguration ユーザーマニュアル』

『Sun Enterprise 10000 SSP ユーザーマニュアル』

名前	sigbcmd - SSP からドメインへのコマンドの送信
形式	<pre>sigbcmd [-svif] [-t retries] [-p proc] cmd</pre> <pre>sigbcmd [-vif] [-t retries] -p proc -r</pre>
注意	このコマンドは購入先の技術者のみが使用します。このコマンドを間違えて使用すると、ドメインの活動を妨げ、予期しないエラーを引き起こします。場合によっては、オペレーティングシステムに重大な障害が発生することもあります。
機能説明	<p>sigbcmd は SSP ユーティリティーであり、次のような処理を実行します：環境変数 <code>SUNW_HOSTNAME</code> によって指定された Sun Enterprise 10000 ドメインへ特定のコマンド (<code>cmd</code>) を送る。メールボックスの状態を表示する (<code>-s</code>)。プロセッサのメールボックスのフラグをリセットして、空白にする (<code>-r</code>)。 <code>cmd</code> が応答を予期している場合、ドメインからその応答を取り出す。</p> <p>sigbcmd はシグニチャーブロックのメールボックスインタフェースを使用してカーネルと通信し、コマンドを実行します。このメールボックスは BBSRAM に常駐しています。</p>
オプション	<p>sigbcmd はコマンドを 1 つだけ実行して終了します。したがって、<code>-r</code>、<code>-s</code>、<code>cmd</code> の各オプションは相互に排他的であり、同時に指定することはできません。同時に複数のオプションを指定すると、<code>-r</code>、<code>-s</code>、<code>cmd</code> の優先順位に従って 1 つだけが使用されます。</p> <p><code>cmd</code> ドメインへ送られるコマンドを表す ASCII 名です。現時点でサポートされているコマンドは、<code>obp</code> (ドメインを OBP に入れます) と <code>panic</code> (ドメインをパニックにします) です。</p> <hr/> <p>注 - オプション <code>-f</code> を使用しない限り、<code>panic</code> または <code>obp</code> コマンドを IDN ドメインに送信することはできません。それらのコマンドを送信するには、まず <code>domain_unlink(1M)</code> を使用してドメインのリンクを解除する必要があります。</p> <hr/> <p><code>-s</code> メールボックスの現在の状態 (<code>flag</code>、<code>cmd</code>、<code>data</code> フィールド) を表示します。データフィールドについては、最初の 32 バイト (<code>SIGB_MBOX_SIZE/2</code>) だけが表示されます。</p> <p><code>-v</code> 冗長 (<code>verbose</code>) モードで実行し、デバッグ情報を出力します。デフォルトは簡潔 (<code>terse</code>) モードです。</p>

- i コマンドを実行するかどうかをユーザーに問い合わせます。デフォルトでは問い合わせません。
- f メールボックスのフラグを無視して、実行を強制します。-f が設定されていない場合は (デフォルト)、メールボックスでなにかが検出されても (すなわち、flag != SIGB_MBOX_EMPTY)、sigbcmd はメールボックスコマンドを書き出しません。-f が指定されていれば、どのような場合でもメールボックスコマンドが書き出されます。このオプションは、IDN に対する実行を強制します。ただし、IDN ドメインがパニックになると、クラスタがアービトレーション停止することがあります。
- r -p *proc* で指定されたターゲットプロセッサのメールボックスのフラグをリセットし、空白にします。
- t *retries* 応答が予期されている場合、メールボックスを読み取り、指定された回数だけその応答を探します。*retries* は整数値です。読み取りと読み取りの間には、1 秒間のポーズが入ります。応答が見つからないときは、エラーメッセージが表示されます。*retries* のデフォルトは 10 であり、探索には約 10 秒かかります。
- p *proc* 指定したプロセッサをコマンドのターゲットとします。デフォルトでは、現在使用できるプロセッサ (通常はブートプロセッサ) が選択されます。

使用例

例 1 OBP へ移動します。

ターゲットのドメインを OBP に入れます。

```
sigbcmd -p 4 obp
```

例 2 コマンドを送信します。

プロセッサ 4 に domain panic コマンドを送ります。ユーザーからの肯定応答を待ってから、コマンドを送ります。

```
sigbcmd -i -p 4 panic
```

環境

環境変数 SUNW_HOSTNAME が該当ドメイン名に設定されている必要があります。

注意事項

このコマンドは、ドメインのオペレーティングシステムの整合性が保たれており、(NMI レベル 15 割り込みを通じて) メールボックスコマンドの受け入れが可能であることを前提とします。

このユーティリティーは権限チェックを行わないことに注意してください。

名前	snmpd - Sun Enterprise 10000 プラットフォームの SNMP プロキシエージェント
形式	snmpd [-h] [-p <i>port</i>] [-X]
注意	このデーモンは手動では実行しないでください。このデーモンは自動的に呼び出されます。snmpd(1M) はプラットフォームタイプのサーバーです。snmpd のインスタンスは、管理および監視されている各プラットフォームについて、SSP スタートアップスクリプトによって定期的に呼び出されます。
機能説明	<p>snmpd(1M) は Sun Enterprise 10000 システム用の SNMPv1 (Simple Network Management Protocol, Version 1) プロキシエージェントです。このエージェントがサービスを提供しているオブジェクトのリストについては、Sun Enterprise 10000 MIB (Management Information Base) 定義ファイルの <code>\$\$SPETC/snmp/Ultra-Enterprise-10000.mib</code> を参照してください。</p> <p>snmpd(1M) は SNMP 要求の <code>set(1)</code>、<code>get(1)</code>、および <code>getnext(1)</code> をサポートします。</p> <p>snmpd(1M) の各インスタンスは、自身の通信ポートを割り当て、<code>machine_server(1M)</code> に登録します。特定の SNMP エージェントと通信しようとする他の SNMP マネージャーも <code>machine_server(1M)</code> を使用して、そのエージェントに割り当てられているポートを見つける必要があります。UDP ポートのいくつかは、この目的のために予約されて <code>/etc/services</code> ファイルに格納されています。</p> <p>SSP に常駐する Sun Enterprise 10000 MIB 情報は、snmpd(1M) によってファイルアクセスデーモン (<code>fad(1M)</code>) を使って提供されます。システムリソースに対応する Sun Enterprise 10000 MIB 情報は、制御ボードサーバー (<code>cbs(1M)</code>) を通じて取り出されます。</p>

オプション

以下のオプションをサポートしています。

- h snmpd(1M) コマンド行オプションの簡単な説明を出力します。
- p *port* 指定した UDP ポートを使って着信 SNMP 要求を受け取ります。-p を指定しない場合は、snmpd(1M) の最初のインスタンスが UDP ポート 5050 を通信用に割り当てます。後続のインスタンスは、任意の UDP ポートを割り当てます。Solstice Enterprise Agent (SEA または snmpdx) が実行されていない場合には、標準ポート番号の 161 でも着信を待機します。ポート割り当て用の `machine_server(1M)` デーモン。
- x snmpd をデーモンにしません。プロセスをフォアグラウンドで継続して実行します。このオプションは、デバッグで使用されます。このオプションを使用しない場合には、snmpd(1M) は、標準出力および標準エラー出力を `/var/tmp/snmpd/snmpd.log` に送信します。

ファイル

以下のファイルがサポートされます。

`$$$PETC/snmp/agt/Ultra-Enterprise-10000.snmpd.cnf`

設定ファイル

`$$$PETC/snmp/Ultra-Enterprise-10000.mib`

Sun Enterprise 10000 MIB 定義ファイル

`$$$PETC/snmp/Ultra-Enterprise-10000.dat`

Sun Enterprise 10000 MIB データファイル

`$$$PVAR/pids/snmpd-platform_name.pid`

プロセス ID ファイル

`$$$PETC/snmp/sea/ssp_snmp.reg`

SSP snmpd(1M) の SEA 登録ファイル

`$$$PETC/snmp/sea/ssp_snmp.rsrc`

SSP snmpd(1M) の Solstice Enterprise Agent リソースファイル

`/var/tmp/snmpd/core`

snmpd(1M) のコアファイル (存在する場合)

`/var/tmp/snmpd/snmpd.log`

標準出力および標準エラー出力を格納するログファイル (出力先が画面にリダイレクトされていない場合)。-X オプションを指定すると、標準出力および標準エラー出力の出力先が画面にリダイレクトされます。

`/var/tmp/snmpd/snmpd.stats`

snmpd の使用状況統計ファイル。このファイルは、snmpd デーモンが SIGUSR1 シグナルを受信したときはいつでも生成されます (たとえば、"kill USR1 551" この 551 は snmpd(1M) のプロセス ID)。このファイルには、snmpd(1M) プロセス開始からの、または SIGUSR1 シグナルを最後に受信してからの取得回数、設定回数、およびトラップ回数が格納されます。

関連項目

`cbs(1M)`, `fad(1M)`, `machine_server(1M)`, `straps(1M)`

名前	ssp_backup - SSP 環境のバックアップ
形式	ssp_backup <i>directory_name</i>
機能説明	<p>ssp_backup は、SSP のオペレーション環境を保守するファイルの cpio(1) アーカイブを作成します。ssp_backup コマンドを使用してメイン SSP をバックアップし、ssp_restore(1M) コマンドを使用してそのアーカイブファイルをスペア SSP に復元すると、メイン SSP とスペア SSP で同じ環境を維持することができます。ssp_backup コマンドは、メイン SSP で実行してください。</p> <p>SSP 環境を変更した場合 (ドメインのシャットダウンなど) は、ssp_backup を実行して SSP のバックアップファイルを最新のものに更新する必要があります。</p> <p>バックアップファイル名は、ssp_backup.cpio です。</p> <p>エラーが発生すると、\$SSPVAR/ssp_backup.out にエラーメッセージが記録されます。</p> <hr/> <p>注 - バックアップファイルのサイズは、非常に大きくなる (20MB 以上) 可能性があります。バックアップファイルのサイズを小さくするには、不必要な SSP ログファイルまたはメッセージファイルを /var/opt/SUNWssp/adm ディレクトリから削除します。</p>
オペランド	<p>以下のオペランドがサポートされています。</p> <p><i>directory_name</i> バックアップファイルが作成されるディレクトリ名。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/ssp_backup.out - ssp_backup ログファイル</p>
関連項目	<p>ssp_restore(1M)</p> <p>『Sun Enterprise 10000 SSP ユーザーマニュアル』</p>

名前	ssp_config - SSP または制御ボードの設定						
形式	ssp_config [cb float]						
機能説明	<p>注 - 手動で SSP または制御ボードのフェイルオーバーを実行するのに ssp_config を使用しないでください。強制的にフェイルオーバーを実行するには setfailover(1M) コマンドを使用してください。</p> <hr/> <p>ssp_config はいくつかの SSP 設定ファイルを設定し、制御ボードに tftpboot 情報を提供します。SSP ソフトウェアをインストールし、SSP をはじめて起動すると、オペレーティングシステムのスタートアップスクリプトによって ssp_config が自動的に実行されます。スーパーユーザーでログインした場合は、ssp_config を手動で実行することもできます。</p> <p>スタートアップスクリプトが ssp_config を実行すると、システム (プラットフォーム) 名、浮動メイン SSP の名前、システム内の制御ボード数、制御ボード 0 の名前、および制御ボード 1 の名前を求めるプロンプトが出されます。さらに主制御ボードの指定を求め、ネームサービスまたはローカルファイルを通じて制御ボードの IP アドレスを自動的に決定しようとしています。自動的な決定がうまくいかない場合は、IP アドレスを指定するためのプロンプトが表示されます。システムが完全に初期設定されるのは、ssp_config が必要な情報をすべて受け取ってからです。</p>						
オプション	<p>以下のオプションをサポートしています。</p> <p>cb cb_config(4) ファイルの制御ボードの設定を変更します。ssp_config cb を実行すると、主制御ボードの指定変更、または制御ボードの交換、追加、削除を促す一連のプロンプトが表示されます。プラットフォーム名を変更することはできません。</p> <p>float 浮動メイン SSP の IP アドレスを設定します。浮動メイン SSP の名前と IP アドレスを問い合わせることができます。</p>						
ファイル	<p>以下のファイルがサポートされます。</p> <table border="0"> <tr> <td>/etc/inetd.conf</td> <td>inet デーモン設定ファイル</td> </tr> <tr> <td>\$\$SSPETC/ssp_env.sh</td> <td>SSP 環境変数ファイル</td> </tr> <tr> <td>\$\$SSPVAR/.ssp_private/cb_config</td> <td>制御ボード設定ファイル</td> </tr> </table>	/etc/inetd.conf	inet デーモン設定ファイル	\$\$SSPETC/ssp_env.sh	SSP 環境変数ファイル	\$\$SSPVAR/.ssp_private/cb_config	制御ボード設定ファイル
/etc/inetd.conf	inet デーモン設定ファイル						
\$\$SSPETC/ssp_env.sh	SSP 環境変数ファイル						
\$\$SSPVAR/.ssp_private/cb_config	制御ボード設定ファイル						

<code>\$\$SSPVAR/.ssp_private/main_ssp_name</code>	浮動メイン SSP の名前と IP アドレスが記録されている
<code>\$\$SSPVAR/etc/platform_name/edd.emc</code>	edd (イベント検出デーモン) イベント監視設定ファイル
<code>\$\$SSPVAR/etc/platform_name/edd.erc</code>	edd (イベント検出デーモン) イベント応答設定ファイル
<code>/tftpboot</code>	制御ボードのイメージの起動ディレクトリ

関連項目

bringup(1M), setfailover(1M), cb_config(4)

名前	ssp_downgrade - 以前のバージョンの SSP に戻す (ダウングレード) よう SSP および Sun Enterprise 10000 システムを準備する
形式	ssp_downgrade [-h] [-v]
機能説明	<p>ssp_downgrade は、以前のバージョンの SSP ソフトウェアに戻すように SSP を準備します。現在のバージョンが SSP 3.5 で、SSP 3.2 を復元する場合のみ、このコマンドを使用してください。SSP 3.3 または SSP 3.4 へ戻す場合は、ssp_downgrade を実行する必要はありません。</p> <hr/> <p>注 - 環境変数 SUNW_HOSTNAME にプラットフォーム名を必ず設定してください。</p> <hr/>
オプション	<p>以下のオプションをサポートしています。</p> <p>-h ssp_downgrade(1M) オプションの簡単な説明を出力します。</p> <p>-v 冗長 (verbose) モードで実行し、デバッグ情報を出力します。</p>
使用例	<p>例 1 以前のバージョンの SSP に戻す準備の実行</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>xf4-ssp:xf4% ssp_downgrade Setting control board 0 Vcc Fan voltage. Control board 0 Vcc Fan voltage set to 5.091013 Setting control board 1 Vcc Fan voltage. Control board 1 Vcc Fan voltage set to 5.093463</pre> </div>
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 以前のバージョンに戻す準備が正常に終了しました。</p> <p>0 以外 エラーが発生しました。</p>
関連項目	ssp_startup(1M)

名前	ssp_restore - SSP 環境の復元
形式	ssp_restore [<i>filename</i>]
機能説明	<p>ssp_restore は、ssp_backup(1M) コマンドを使用して作成したバックアップファイルから SSP のオペレーション環境を復元します。ssp_restore を使用すると、メイン SSP とスペア SSP で同じ環境を維持することができます。そのためには、ssp_backup(1M) を使用してメイン SSP のバックアップを作成し、そのアーカイブファイルを ssp_restore を使用してスペア SSP に復元します。SSP ソフトウェアが新しいディスクにインストールされた後に SSP 環境を復元する場合にも ssp_restore を使用することができます。</p> <p>/var/opt/SUNWssp/.ssp_private/ssp_to_domain_hosts など、必要な構成ファイルは、すべて自動的に変換されます。</p> <p>エラーが発生すると、\$SSPVAR/ssp_restore.out にエラーメッセージが記録されます。</p> <hr/> <p>注 - バックアップファイルの作成後、メイン SSP の環境が変更された場合 (ドメインのシャットダウンなど) は、ssp_backup(1M) を実行して SSP のバックアップファイルを最新のものに更新する必要があります。この新しいバックアップファイルで ssp_restore コマンドを実行することにより、メイン SSP とスペア SSP の両方の環境を同一に保つ必要があります。</p> <hr/>
オペランド	<p>以下のオペランドがサポートされています。</p> <p><i>filename</i> ssp_backup(1M) によって作成されたバックアップファイルの名前。指定されたファイルがカレントディレクトリにない場合には、ファイルのフルパス名を記述する必要があります。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/ssp_restore.out ssp_restore ログファイル</p>
関連項目	<p>ssp_backup(1M)</p> <p>『Sun Enterprise 10000 SSP ユーザーマニュアル』</p>

名前	ssp_startup - SSP デーモンの起動	
形式	ssp_startup	
機能説明	<p>注 - このコマンドはコマンド行から実行しないでください。通常、ssp_startup スクリプトは、/etc/init.d/ssp で定義された再起動のたびに自動的に実行され、/etc/rc2.d/S99ssp にリンクされます。</p> <hr/> <p>ssp_startup スクリプトは、現行の設定に応じた適切な順序で、以下の Sun Enterprise 10000 システムデーモンを実行します (下記は一例で、実際の順序は下記と異なる場合があります)。</p> <ul style="list-style-type: none"> ■ edd(1M) ■ cbs(1M) ■ snmpd(1M) ■ machine_server(1M) ■ fad(1M) ■ fod(1M) ■ datasyncd(1M) ■ straps(1M) ■ obp_helper(1M) ■ netcon_server(1M) <p>ssp_startup は、必要に応じてこれらのデーモンを監視し再起動します。</p> <p>ssp_startup が完了すると、以下のメッセージを \$SSPLOGGER/messages ファイルに記述します。</p> <pre>Startup of SSP as MAIN complete</pre> <p>上のメッセージは、どちらの SSP (MAIN = メイン、SPARE = スペア) が起動プロセスを完了したかを示します。</p>	
ファイル	以下のファイルがサポートされます。	
	\$SSPETC/ssp_startup.main	メイン SSP での初回のスタートアップ時にどのプログラムを起動するかを決定します。
	\$SSPETC/ssp_startup.remote	スペア SSP での初回のスタートアップ時にどのプログラムを起動するかを決定します。

<code>\$SSPETC/ssp_startup.restart_main</code>	必要に応じてメイン SSP 上で監視・再起動するプ ログラムを決定します。
<code>\$SSPETC/ssp_startup.restart_spare</code>	必要に応じてスペア SSP 上で監視・再起動するプ ログラムを決定します。

名前	ssp_unconfig - SSP の設定の解除
形式	ssp_unconfig
機能説明	<p>注 - ssp_unconfig は、システムの出荷直前に Sun Microsystems が使用します。ユーザーがこのコマンドを使用する場合は、Sun Enterprise 10000 サーバーに関する詳しい知識が必要です。いずれにしても、使用の前には、サービスアドバイザに相談してください。</p> <hr/> <p>ssp_unconfig を使用するのには、新しい環境で SSP を再設定する場合、あるいはホスト名、IP アドレス、またはプラットフォーム名を変更する場合があります。ssp_unconfig は、シリアル ID を使用して eeprom.image ファイルが初めて作成されるドメイン名の入力を促します。sys_id(1M) の -s オプションを参照してください。</p> <hr/> <p>注 - このコマンドを実行するときには、ドメインをすべて停止し、domain_remove(1M) を使って削除しておいてください。また eeprom.image ファイルを保存してください。これらのファイルは、ドメインの再作成の際に対応するドメインに復元されないことがあります。</p> <hr/>
ファイル	<p>以下のファイルがサポートされます。</p> <pre> \$SSPETC/ssp_env.sh \$SSPVAR/.ssp_private/cb_config \$SSPVAR/.ssp_private/domain_config \$SSPVAR/.ssp_private/domain_history \$SSPVAR/.ssp_private/main_ssp_name \$SSPVAR/.ssp_private/ssp_to_domain_hosts \$SSPVAR/etc/platform_name/edd.erc \$SSPVAR/etc/platform_name/edd.emc /.SSP_DEFAULTS /tftpboot </pre>
関連項目	<p>ssp_config(1M), domain_remove(1M)</p> <p>SunOS リファレンスマニュアルの halt(1M), shutdown(1M)</p>

名前	straps - SNMP トラップシンクサーバー
形式	straps
注意	このコマンドは手動では実行しないでください。このコマンドは SSP スタートアップスクリプトによって自動的に呼び出され、必要に応じて再起動されます。
機能説明	<p>straps サーバーは、SNMP トラップポートから着信トラップメッセージを受け取り、接続されている全クライアントに転送します。</p> <p>SNMP トラップポートは UDP ポート番号 5051 です。このポートは特権ポートであり、一度しか開くことができません。したがって、straps は、SSP 上の複数の SNMP マネージャータイプのアプリケーションにサービスを提供する必要があります。SNMP トラップを受け取る必要があるクライアントには、原則として straps サーバーとの接続が確立されている必要があります。この接続は、AF_UNIX ドメインストリームソケットへ接続し、SNMP トラップ パケットを raw バイナリ形式で取得することによって行われます。</p> <p>各 SSP が一度に実行できる straps のインスタンスは 1 つだけです。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPETC/snmp/straps - ストリームソケットデバイスファイル</p> <p>\$SSPVAR/pids/straps.pid - プロセス ID ファイル</p>
関連項目	snmpd(1M)

名前	sys_clock - システムクロック周波数の表示・変更・設定
形式	sys_clock sys_clock -m ctrl_board_num [-i ic_freq] [-p proc_multiple] [-s] sys_clock [-i ic_freq] [-p proc_multiple] [-s]
注意	<p>コマンド行でこのオプションを実行する場合には、専門的な知識を持つユーザーでない限り、オプションを付けずにデフォルトの形式のまま実行してください。オプションを付けた場合、このコマンドはターゲット周波数のリセットまたは変更を試みます。こうした試みは、システムのパフォーマンスの低下またはクラッシュの原因となります。</p> <hr/> <p>注 - <code>sys_clock</code> コマンドに <code>-s</code> オプションを指定し、クロック比率またはクロック周波数の設定を変更すると、動作中のドメインがすべてアービトレーション停止します。</p> <hr/>
機能説明	<p>コマンド行オプションを付けずに <code>sys_clock</code> を実行すると、ターゲットの周波数と実際の周波数 (インターコネクと JTAG のクロック周波数) の両方が表示され、さらにプロセッサ対インターコネクのクロック比率が表示されます。</p> <p>SSP パッケージは一連のターゲットクロック周波数が組み込まれて配布されています。ターゲット周波数は Sun Enterprise 10000 システムのパフォーマンスを最大化することを目的としており、snmpd(1M) エージェントによって維持されます。通常、センタープレーンサポートボードとシステムボードは、2つの制御ボードのうちのいずれか1つからクロック周波数を受け取ります。デフォルトのソースクロックは主制御ボードです。</p> <p><code>sys_clock</code> は SSP の起動時に自動的に実行されます。</p>

オプション

以下のオプションをサポートしています。

注 `-i` オプションまたは `-p` オプションを付けて `sys_clock` を実行した場合、`snmpd(1M)` エージェントが維持しているターゲット設定が変更されます。この設定は、次回 `sys_clock -s` を実行したときに Sun Enterprise 10000 システムに渡されます。`-i` または `-p` を使用して Sun Enterprise 10000 システムの設定を即座に変更するには、コマンド行に `-s` を付け加えてください。

- | | |
|--------------------------------|--|
| <code>-i ic_freq</code> | インターコネク트의ターゲットクロック周波数を、指定した値 (ヘルツ) に変更します。有効値は 40000000(40.0Mhz) ~ 120000000(120.0Mhz) です。 |
| <code>-m ctrl_board_num</code> | システムクロックを供給している制御ボードの番号でボードをプログラミングします。 |
| <code>-p proc_multiple</code> | プロセッサ対インターコネク트의ターゲットクロック比率を、指定した値に設定します。値は two-to-one(2 対 1)、three-to-one(3 対 1)、three-to-two(3 対 2)、または five-to-two(5 対 2) のいずれかです。 |
| <code>-s</code> | <p><code>-m</code> オプションと同様、システムクロックを供給している制御ボードの番号でボードをプログラミングします。しかし、<code>-m</code> とは異なり、ハードウェアのクロック周波数を、<code>snmpd(1M)</code> エージェントが維持しているターゲット周波数に設定します。このオプションを使用する前に、必ず、「注意」をお読みください。</p> <p><code>sys_clock -s</code> は、Sun Enterprise 10000 システムの電源を投入するたびに自動的に実行されます。</p> |

使用例

例 1 ターゲットのクロック周波数と実際のクロック周波数を表示します。

```
% sys_clock
Current Clock Frequencies:
-----
Interconnect:83.25 Mhz
Processor:333.01 Mhz
JTAG:5.20 Mhz

Targeted Clock Frequencies:
-----
Interconnect:83.33 Mhz
Proc Clock Ratio: three-to-two
JTAG:5.00 Mhz
```

注 - JTAG クロック周波数は、JTAG 周波数の表示をサポートする制御ボードでのみ表示されます。

例 2 ターゲットのクロック周波数を変更します。

```
% sys_clock -i 83333333 -p two-to-one
```

例 3 システムのクロック周波数を設定します。

```
% sys_clock -s
```

エラー

Error: Failed to get SSP host name for domain xf4

snmpd(1M) エージェントが実行されていない可能性があります。このメッセージは、構成されているドメインがなく、環境変数 SUNW_HOSTNAME がドメイン名に設定されていない場合に返されることがあります。SUNW_HOSTNAME をプラットフォーム名に設定してみてください。

ファイル

以下のファイルがサポートされます。

```
$SSPETC/snmp/Ultra-Enterprise-10000.mib
Sun Enterprise 10000 MIB 定義ファイル
```

関連項目

cbs(1M), snmpd(1M), ssp_resource(4)

名前	sys_id - SSP の IDPROM に設定されたシステム ID の表示または変更
形式	<pre> sys_id sys_id -d [-f filename] sys_id -x [-f filename] sys_id -m [yy] yymmddhhmm [-f filename] sys_id -k key -s serial_no [-f filename] sys_id -k key -h host_id [-f filename] </pre>
機能説明	<p>sys_id は、Sun Enterprise 10000 SSP の IDPROM に設定されたホスト ID と Ethernet 番号を表示または変更します。変更された設定は、<code>eeprom.image</code> ファイルに保存されます。</p> <p>システム ID (マシンのシリアル番号、ホスト ID、および Ethernet 番号) の管理情報は、EEPROM イメージの IDPROM 部分として SSP 上のファイルに格納されます。</p> <p>EEPROM イメージの IDPROM 部分は勝手に変更されないように暗号化されています。ただし、システム ID を変更するために、<code>eeprom.image</code> ファイル内の暗号化された ID を変更することは可能です。これを行うには、まずサンから <i>key</i> を入手する必要があります。このキーによって、指定されたマシンシリアル番号以外の値にシステム ID を変更することが防止されます。</p> <p>他と重複しないように、Sun Enterprise 10000 システムのホスト ID と Ethernet 番号は FOA/シリアル番号から取り出されます。Sun Enterprise 10000 システムのシリアル番号は、SPARC International からサンに割り当てられたブロック番号に基づいています。</p>
システム ID	<p>sys_id コマンドは、ホスト名と環境変数 <code>SUNW_HOSTNAME</code> がサンから提供された値と一致する SSP 上で実行してください (キーが与えられた場合)。さらに、(IDPROM から取得した) 既存のホスト ID が <code>-s serial_number</code> または <code>-h host-id</code> と一致している必要もあります。</p>
オプション	<p>引数を付けずに <code>sys_id</code> を実行した場合は、<code>eeprom.image</code> ファイル (デフォルトの場所に格納されています) に設定されているホスト ID が 10 進数で表示されます。</p> <p><code>-d</code> IDPROM のフィールドをすべて表示します。製造年月日は UTC (世界協定時刻の略) で表示されます。</p>

- m [yy]yymmddhhmm** IDPROM の製造年月日と時刻を、指定した年 (最初の 2 桁はオプション)、月、日、時間、分に設定します。最初の *yy* が指定されておらず、年に 69 ~ 99 の 2 桁の数字が指定されている場合には、**sys_id** は 1969 ~ 1999 年と解釈します。最初の *yy* が指定されておらず、年に 00 ~ 68 が指定されている場合には、2000 ~ 2068 年と解釈します。
sys_id -m を実行できるのは、1 回だけです。
- f filename** EEPROM イメージのデフォルトの格納場所とファイル名を *filename* に変更します。ファイルの読み書きが実行中であるかどうかに関わらず、**-f** は指定された場所を使用します。
EEPROM イメージファイルのデフォルトの場所と名前は、`$$SSPVAR/etc/platform_name/$$SUNW_HOSTNAME/eprom.image` です。
- k key** システム ID を変更します。このオプションを付けて **sys_id** を実行するには、あらかじめサンからキーを入手しておく必要があります。コマンドの実行場所と方法は、キーによって制限されます。**-k** オプションは、追加のドメインホスト (複数のドメイン) 用に ID を取得または変更するための唯一の方法です。追加のドメインには、0x80A66000 ~ 0x80A66DFF および 0x80A67000 ~ 0x80A6FFFF の範囲の値を使用してください。
このオプションを指定する場合は、**-s** または **-h** も一緒に指定する必要があります。
- s serial_no** システム ID を変更します。**-s** は IDPROM 内のホスト ID エントリのシリアル番号部分を指定されたシリアル番号に変更します。この番号は、サンから入手したものである必要があります。シリアル番号は、10 進法または 16 進法で指定することができます。16 進法で指定する場合は、0x または 0X の接頭辞を付けてください。シリアル番号の有効な範囲は、0xA65000 ~ 0xA65FFF、0xA66E00 ~ 0xA66FFF、および 0xA70000 ~ 0xA70FFF です。

-h *host*

ホスト ID を変更します。-h は、IDPROM 内のホスト ID エントリのホスト ID 部分を指定されたホスト ID に変更します。この番号は、サンから入手したものである必要があります。ホスト ID は、10 進法または 16 進法で指定することができます。16 進法で指定する場合は、0x または 0X の接頭辞を付けてください。ホスト ID の有効な範囲は 0x80A65000 ~ 0x80A70FFF です。

-x

シリアル番号を 16 進法で表示します。

名前	thermcal - サーミスターのキャリブレーションの読み取りまたは書き込み
形式	thermcal [-r -w] -b sb -n <i>sb_number</i> thermcal [-r -w] -b cp -n <i>cp_number</i>
注意	このコマンドは実行しないでください。これはサンの技術者および購入先の技術者のみが使用するコマンドです。
機能説明	thermcal は、システムボードおよびセンタープレーン上の ASIC 用 EEPROM に格納されているサーミスター値を表示します。
オプション	以下のオプションをサポートしています。 -b sb -n <i>sb_number</i> 指定したシステムボードでの読み書きを行います。 <i>sb_number</i> は 0 ~ 15 の整数であり、システムボード番号を表します。 -b cp -n <i>cp_number</i> 指定したセンタープレーンでの読み書きを行います。 <i>cp_number</i> は 0 または 1 の整数であり、センタープレーン番号を表します。 -r サーミスターのキャリブレーションを読み取ります。これはデフォルトの設定です。 -w サーミスターのキャリブレーションを書き込みます。このオプションを使用するのは、サンの製造技術者のみです。ボードを正しくキャリブレートするには、まずボードの電源を切断し (power(1M) を参照)、少なくとも 30 分間アイドル状態で放置してから、thermcal -w を実行します。ボードの電源は、thermcal がキャリブレーションを実行するときに、自動的に投入されます。

名前	thermcal_config - SSP サーミスターキャリブレーションのデータファイルの作成
形式	<pre>thermcal_config thermcal_config -u thermcal_config -b sb -z n thermcal_config -b csb -z n</pre>
注意	このコマンドは、edd(1M) により自動的に実行されます。通常、コマンド行では実行しません。
機能説明	オプションを指定しない場合は、thermcal_config は SSP 設定ファイル thermcaldata.tcl を作成します。このファイルには、システムボードとセンタープレーン上の ASIC コンポーネントに使用する (非精密、分離式) サーミスターに関するキャリブレーション値が格納されます。SSP はこのファイルを使用して、ASIC の温度をチェックします。
オプション	<p>以下のオプションをサポートしています。</p> <pre>-u</pre> <p>すべての変更に対して、既存の thermcaldata.tcl ファイルを更新します。</p> <pre>-b sb -z n</pre> <p>指定したシステムボードに変更があった場合にのみ更新します。他の変更はすべて無視します。<i>n</i> は、0 ~ 15 まで整数であり、システムボード番号を表わします。</p> <pre>-b csb -z n</pre> <p>指定したセンタープレーンサポートボードに変更があった場合にのみ更新します。他の変更はすべて無視します。<i>n</i> は 0 または 1 の整数であり、センタープレーンサポート番号を表わします。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <pre>\$SSPVAR/etc/platformname/cbobjs/thermcaldata.tcl</pre>
関連項目	edd(1M)

第2章

ファイル形式

名前	Intro - SSP 3.5 ファイル、その他
機能説明	このセクションには、SSP 環境で実行できる各種のファイルやスクリプトなどに関するマニュアルページが収録されています。
コマンド一覧	以下のコマンドがサポートされます。
	blacklist 起動されないシステムリソースのリスト
	cb_config SSP が管理するマシンのリスト
	cb_port cbe 用の通信ポートのリスト
	domain_config 設定済みドメインのリスト・記述
	domain_history 削除されたドメインのリスト・記述
	edd.emc イベント監視設定ファイル
	edd.erc イベント応答設定ファイル
	fad_files ファイルアクセスデーモンファイル
	postrc hpost 属性ファイル
	redlist 操作されないシステムリソースのリスト
	ssp_resource SSP プロセスのリソースファイル
	ssp_to_domain_hosts ホスト名とドメイン名のファイル

名前	blacklist - 起動されないシステムリソースのリスト
機能説明	<p>\$SSPVAR/etc/<i>platform_name</i>/blacklist は ASCII ファイルです。システム管理者 (またはスーパーユーザー) はこのファイルを使用して、ホストシステムの設定を SSP から制限することができます。このファイルには、POST が起動時に使用することができないコンポーネントがリストされています。POST は blacklist ファイルを読み取ってから、システムを起動するための準備を行い、テストに成功したコンポーネントだけを OBP に渡します。blacklist にリストされているコンポーネントは、OBP には渡されません。</p> <p>blacklist ファイルには、以下の規則が適用されます。</p> <ul style="list-style-type: none"> ■ キーワードでは大文字と小文字が区別されません。 ■ # で始まる行部分はコメントとみなされます。 ■ 0x で始まる場合を除き、数字はすべて 10 進数で指定されます。0x で始まる数字は 16 進数です。ただし、[a - f] または [A - F] のいずれかで指定されたボード番号は 16 進数とみなされます。 ■ 各行にはキーワードが 1 つだけ存在します。 ■ 同じキーワードを複数の行で使用することができます。 ■ 各キーワードには 1 つまたは複数の引数を使用することができます。引数は整数値で表され、各整数値はピリオドで区切られます。
キーワード	<p>下記のすべての値の範囲は境界値を含みます。</p> <p>-sysbd <i>board</i></p> <p>指定したシステムボードをテストまたは設定しません。<i>board</i> は 0 ~ 15 の整数です。</p> <p>-proc <i>board.pmod</i></p> <p>指定したシステムボード内の指定したプロセッサをテストまたは設定しません。<i>board</i> は 0 ~ 15 の整数、<i>pmod</i> は 0 ~ 3 の整数です。</p> <p>-abus <i>abus</i></p> <p>指定したアドレスバスをテストまたは設定しません。<i>abus</i> は 0 ~ 3 の整数です。すべてのシステムボードについて、対応する CIC ASIC がブラックリストに入れられます。下記のキーワード <i>cic</i> を参照してください。</p> <p>-dbus <i>dbus</i></p> <p>144 ビットのデータルータのうち、指定した 72 ビット ハーフをテストまたは設定しません。<i>dbus</i> は 0 または 1 の整数です。すべてのシステムボード上にあるローカルデータルータについて、対応するハーフ部分がブラックリストに入れられます。下記のキーワード <i>ldpath</i> を参照してください。</p>

-ioc *board.ioctl*

指定したシステムボード上の指定した入出力コントローラをテストまたは設定しません。*board* は 0 ~ 15 の整数、*ioctl* は 0 または 1 です。

-scard *board.ioctl.slot*

指定したシステムボード上の指定した入出力コントローラ内にある指定した入出力アダプタをテストまたは設定しません。*board* は 0 ~ 15 の整数、*ioctl* は 0 または 1、*slot* は 0 ~ 3 の整数です。

注 - *scard* の 0 ~ 3 の範囲の整数は、ブラックリストファイルで構文的に正しい値を表わします。この値により入出力モジュールを将来追加することが可能になります。サポートされるモジュールタイプは、現時点では入出力コントローラ 1 台に対し、2 枚の SBus カード、1 枚の PCI カードのいずれかです。ブラックリスト内の存在しない *scard* は単に無視されます。

-mem *board*

指定したシステムボード上のメモリーをテストまたは設定しません。*board* は 0 ~ 15 の整数です。

-mgroup *board.group*

指定したシステムボード上の、指定した DIMM メモリーのグループをテストまたは設定しません。*board* は 0 ~ 15 の整数、*group* は 0 ~ 3 の整数です。

-mlimit *board.group.MBytes*

指定したシステムボード上の指定した DIMM グループのメモリー設定を指定した値に制限します (実際の値よりも小さく制限します)。*board* は 0 ~ 15 の整数、*group* は 0 ~ 3 の整数、*MBytes* は 64 または 256 です。このキーワードはベンチマークを実行するときには有用です。たとえば、実際には 32 MB の DIMM (256 MB グループ) を搭載しているマシンで、8 MB の DIMM (64 MB グループ) のメモリーをシミュレートするような場合です。256 MB のメモリーグループを搭載した 1 枚のボードと、それぞれ 64 MB のメモリーグループを搭載した 4 枚のボードとでは動作が異なるため、こうした詳細な制御が必要になります。

-cplane *half_centerplane*

Sun Enterprise 10000 のセンタープレーンのうち、指定したハーフ部分をテストまたは設定しません。センタープレーンハーフには、2つのアドレスバスと72ビットのグローバルデータルータが備わっています。*half_centerplane* は0または1です。このコマンドの意味は、当該センタープレーンハーフのバスに対して *abus* と *dbus* の両方のコマンドを指定した場合と同じです。

pc *board.pc*

指定したシステムボード上の指定したポートコントローラ ASIC をテストまたは設定しません。*board* は0～15の整数、*pc* は0、1、2のいずれかです。

xdb *board.xdb*

指定したシステムボード上の指定したデータバッファ ASIC をテストまたは設定しません。*board* は0～15の整数、*xdb* は0～3の整数です。

cic *board.cic*

指定したシステムボード上の指定したコヒーレントインタフェースコントローラ ASIC をテストまたは設定しません。*board* は0～15の整数、*cic* は0～3の整数です (*cic* は該当するボードのアドレスバスに対応します)。

ldpath *board.dbus*

指定したシステムボード上の144ビットのローカルデータルータのうち、指定した72ビットハーフをテストまたは設定しません。*board* は0～15の整数、*dbus* は0または1の整数です。

使用例

例1 ブラックリストファイルの例

```
# Sun Microsystems, Inc.
sysbd 3 5 0xA # システムボード 3、5、10 を使用不可にする
sysbd 3 5 A # システムボード 3、5、10 を使用不可にする
PROC 4.0 6.2 # システムボード 4 のプロセッサ 0 および
# システムボード 6 のプロセッサ 2 を使用不可にする
ScarD 3.0.1 # システムボード 3 の入出力コントローラ 0 に
# 装着されている入出力アダプタ 1 を使用不可にする
mem 2 # システムボード 2 のメモリーをすべて使用不可にする
mlimit 0xE.2.64
# システムボード 14 の DIMM グループ 2 の
# 使用を 64 MB に制限する
cIc 1.2 # システムボード 1 の CIC ASIC 2 を使用不可にする
```

関連項目

hpost(1M), **redlist(4)**

名前	cb_config - SSP が管理するマシンのリスト
注意	このファイルは、手動では編集しないでください。このファイルは、ドメイン管理用のツールおよびコマンドにより自動的に保守されます。制御ボードを変更する場合は、オプション cb を指定して ssp_config(1M) を実行してください。
機能説明	<p>\$SSPVAR/.ssp_private/cb_config ファイルは、SSP が管理し、主制御ボードと代替制御ボードの両方の名前が記述されているマシンを特定します。このファイルの各行がそれぞれ 1 台のマシンを表し、各行には次のようなコロンで区切られたフィールドがあります (以下の構文説明では、見やすくするために項目と項目の間にスペースを入れています、実際にはスペースは使用しません)。</p> <pre>platform_name : platform_type : cb0_hostname : status0 : cb1_hostname : status1</pre> <p>上記で、</p> <p><i>platform_name</i> Enterprise 10000 マシンの名前です。これは論理名であり、ブート可能なドメインを表すものではありません。</p> <p><i>platform_type</i> マシンのプラットフォームタイプです。現行リリースでは、<i>platform_type</i> は Ultra-Enterprise-10000 だけです。</p> <p><i>cb0_hostname</i> マシンのスロット 0 に装着された制御ボードに割り当てられているホスト名です。</p> <p><i>status0</i> <i>cb0</i> が主制御ボードであるかどうかを示す値です。<i>status0</i> の値が P であれば <i>cb0</i> が主制御ボードであり、それ以外の値であれば主制御ボードではありません。</p> <p><i>cb1_hostname</i> マシンのスロット 1 に装着された制御ボードに割り当てられているホスト名です。</p> <p><i>status1</i> <i>cb1</i> が主制御ボードであるかどうかを示す値です。<i>status0</i> の値が P であれば、<i>status1</i> フィールドは空白になっています。<i>status0</i> の値が P 以外であれば、<i>status1</i> が P になります。制御ボードにホスト名が指定されていない場合、そのボードの <i>status</i> フィールドは空白になります。</p>

使用例

例 1 cb_config ファイルの例

```
xf2:Ultra-Enterprise-10000:xf2-cb0:P:xf2-cb1:
```

プラットフォーム xf2 で、xf2-cb0 はスロット 0 の制御ボードであり、主制御ボードとして設定されています。代替制御ボード xf2-cb1 はスロット 1 に装着されています。

ファイル

以下のファイルがサポートされます。

```
$SSPVAR/.ssp_private/cb_config
```

関連項目

cbs(1M), **ssp_config(1M)**, **domain_config(4)**

名前	cb_port - cbe 用の通信ポートのリスト
機能説明	<p>\$SSPVAR/.ssp_private/cb_port ファイルは、制御ボードエグゼクティブ (cbe) 用の入力と出力の通信ポートの番号を特定します。cbe は制御ボード上で実行されるプログラムであり、JTAG やその他の要求に対して機能します。CBE と通信する SSP プログラムは制御ボードサーバー cbs(1M) だけです。</p> <p>このファイルでは、入力と出力のポートの番号がそれぞれ 1 行で示されます。</p>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/.ssp_private/cb_port</p>
関連項目	cbs(1M)

名前	domain_config - 設定済みドメインのリスト・記述
注意	このファイルは手動では編集しないでください。このファイルはドメイン管理用のツールとコマンドによって自動的に保守されます。このファイルを手動で編集すると、システムに重大な障害を引き起こす場合があります。
機能説明	<p>domain_config ファイルには、domain_create(1M) を使って作成した設定済みのドメインが各行に 1 つずつ示されます。cbs(1M) はこのファイルを使用して、管理すべきプラットフォームを判定します。このファイルの内容を表示するには、domain_status(1M) コマンドを使用します。</p> <p>リストされるドメインの数に制限はありません。domain_config ファイルの各行は、次のような形式になっています。</p> <p><i>domain_name</i> : <i>platform_type</i> : <i>platform_name</i> : <i>os_version</i> : <i>idn_info</i> : <i>sysbds</i></p> <p>上記で、</p> <p><i>domain_name</i> 設定済みドメインの名前です。ドメイン名は、そのドメインの設定ファイルが属する SSP によってサポートされている全物理プラットフォームにおいて、一意の名前である必要があります。</p> <p><i>platform_type</i> ドメインが属するシステムのタイプです。現在のところ、<i>platform_type</i> は Ultra-Enterprise-10000 だけです。</p> <p><i>platform_name</i> ドメインが属する Sun Enterprise 10000 システムの名前です。</p> <p><i>os_version</i> 現在ドメインが実行している、Sun Enterprise 10000 システムの Solaris オペレーティングシステムのバージョンです (たとえば、2.5.1)。</p> <p><i>idn_info</i> InterDomain Networks ソフトウェアが使用する識別情報。この情報により、どのドメインが IDN ベースのネットワークのメンバーかが判別されます。</p> <p><i>sysbds</i> ドメインに割り当てられているシステムボードです。各ボード番号がそれぞれスペースで区切られて示されます。</p>

使用例

例 1 domain_config ファイルの例

```
marvin:Ultra-Enterprise-10000:production:2.4.3.9:-1.0:2 4 6 8 10
nebula:Ultra-Enterprise-10000:test:2.5.0.1:-1.0:12 14 15
```

ファイル

以下のファイルがサポートされます。

`$SSPVAR/.ssp_private/domain_config`

関連項目

`cbs(1M)`, `domain_create(1M)`, `domain_remove(1M)`,
`domain_status(1M)`

『Sun Enterprise 10000 Dynamic Reconfiguration リファレンスマニュアル』
の `drain(1M)`, `init_attach(1M)`

名前	domain_history - 削除されたドメインのリスト・記述												
注意	このファイルは手動では編集しないでください。このファイルは、ドメイン管理用のツールとコマンドによって自動的に保守されます。												
機能説明	<p>domain_history ファイルには、domain_remove(1M) コマンドによってシステムから削除されたドメインが各行に 1 つずつ示されます。domain_create(1M) コマンドはこのファイルを調べて、作成しようとするドメインが以前に存在していたかどうかを判別します。</p> <p>現在設定されているドメインについては、domain_config(4) を参照してください。</p> <p>domain_history ファイルの各行は、次のような形式になっています。</p> <pre>domain_name : platform_type : platform_name : os_version : idn_info : sysbds</pre> <p>上記で、</p> <table border="0"> <tr> <td style="padding-right: 20px;"><i>domain_name</i></td> <td>削除されたドメインの名前です。</td> </tr> <tr> <td><i>platform_type</i></td> <td>削除されたドメインが属していたシステムのタイプです。現在のところ、<i>platform_type</i> は Ultra-Enterprise-10000 だけです。</td> </tr> <tr> <td><i>platform_name</i></td> <td>削除されたドメインが属していた Sun Enterprise 10000 システムの名前です。</td> </tr> <tr> <td><i>os_version</i></td> <td>削除されたドメインが実行していた、Sun Enterprise 10000 システムの Solaris オペレーティングシステムのバージョンです (5.7 など)。</td> </tr> <tr> <td><i>idn_info</i></td> <td>未使用。</td> </tr> <tr> <td><i>sysbds</i></td> <td>削除されたドメインに割り当てられていたシステムボードです。各ボード番号がスペースで区切られて表示されます。</td> </tr> </table>	<i>domain_name</i>	削除されたドメインの名前です。	<i>platform_type</i>	削除されたドメインが属していたシステムのタイプです。現在のところ、 <i>platform_type</i> は Ultra-Enterprise-10000 だけです。	<i>platform_name</i>	削除されたドメインが属していた Sun Enterprise 10000 システムの名前です。	<i>os_version</i>	削除されたドメインが実行していた、Sun Enterprise 10000 システムの Solaris オペレーティングシステムのバージョンです (5.7 など)。	<i>idn_info</i>	未使用。	<i>sysbds</i>	削除されたドメインに割り当てられていたシステムボードです。各ボード番号がスペースで区切られて表示されます。
<i>domain_name</i>	削除されたドメインの名前です。												
<i>platform_type</i>	削除されたドメインが属していたシステムのタイプです。現在のところ、 <i>platform_type</i> は Ultra-Enterprise-10000 だけです。												
<i>platform_name</i>	削除されたドメインが属していた Sun Enterprise 10000 システムの名前です。												
<i>os_version</i>	削除されたドメインが実行していた、Sun Enterprise 10000 システムの Solaris オペレーティングシステムのバージョンです (5.7 など)。												
<i>idn_info</i>	未使用。												
<i>sysbds</i>	削除されたドメインに割り当てられていたシステムボードです。各ボード番号がスペースで区切られて表示されます。												
使用例	<p>例 1 domain_history(4) ファイルの例</p> <pre>nebula : Ultra-Enterprise-10000 : igor : 5.6 : 0.0 : 4 6 8</pre>												
ファイル	<p>以下のファイルがサポートされます。</p> <pre>\$SSPVAR/.ssp_private/domain_history</pre>												
関連項目	domain_create(1M), domain_remove(1M), domain_config(4)												

名前	edd.emc - イベント監視設定ファイル				
機能説明	<p>edd.emc ファイルは ASCII ファイルです。システムが特定のイベントをどのように監視するかは、このファイルによって指定されます。システムごとに必ず 1 つの EMC ファイルがあります。</p> <p>EMC ファイルは、ssp_config(1M) コマンドの実行時に、テンプレートファイルに基づいて作成されます。テンプレートファイルは、<code>\$SSPVAR/.ssp_private/templates/Ultra-Enterprise-10000</code> に <code>edd.emc</code> という名前で格納されています。</p> <p>EMC ファイルの各行は、それぞれ次のような形式になっています。</p> <pre>event_type : load_event_monit</pre> <p>上記で</p> <table border="0"> <tr> <td data-bbox="503 835 747 873"><i>event_type</i></td> <td data-bbox="747 835 1352 894">イベントの種類を示すニモニック (名前の文字列) です。下記の「使用例」を参照してください。</td> </tr> <tr> <td data-bbox="503 911 747 949"><i>load_event_monit</i></td> <td data-bbox="747 911 1352 1062">有効か無効かのキーワードです。有効 (enabled) にすると、CBE 上のイベントタイプに応じたイベント監視スクリプトが読み込まれます。無効 (disabled) にすると、イベント監視スクリプトは読み込まれません。</td> </tr> </table> <p>各フィールドはコロンで区切られます。コロンの左右にはスペースを入れても入れなくてもかまいません。ハッシュ記号 (#) で始まる行はコメントであり、構文解析の対象とはなりません。edd.emc ファイルの内容は次のように設定されています。</p> <ul style="list-style-type: none"> システムボードの温度イベント システムボードの電圧イベント 制御ボードの温度イベント 制御ボードの電圧イベント センタープレーンの温度イベント センタープレーンの電圧イベント センタープレーンサポートボードの温度イベント センタープレーンサポートボードの電圧イベント ホスト回復イベント その他のイベント 	<i>event_type</i>	イベントの種類を示すニモニック (名前の文字列) です。下記の「使用例」を参照してください。	<i>load_event_monit</i>	有効か無効かのキーワードです。有効 (enabled) にすると、 CBE 上のイベントタイプに応じたイベント監視スクリプトが読み込まれます。無効 (disabled) にすると、イベント監視スクリプトは読み込まれません。
<i>event_type</i>	イベントの種類を示すニモニック (名前の文字列) です。下記の「使用例」を参照してください。				
<i>load_event_monit</i>	有効か無効かのキーワードです。有効 (enabled) にすると、 CBE 上のイベントタイプに応じたイベント監視スクリプトが読み込まれます。無効 (disabled) にすると、イベント監視スクリプトは読み込まれません。				

システムボードの 温度イベント	<pre>sys_brd_temp_norm : enabled # ボードの温度が過熱状態から通常温度に戻りました。 sys_brd_temp_warn : enabled # システムボードの温度が警告値を超過しました。 sys_brd_temp_max : enabled # システムボードの温度の最大値を超過しました。 sys_brd_temp_911 : enabled # システムボードの温度がしきい値 911 を超過しました。 sys_brd_temp_bad : enabled # システムボードの温度を読み取ることができません。 sys_brd_temp_change : enabled # システムボードの温度があらかじめ定められた値だけ変化しました。</pre>
システムボードの 電圧イベント	<pre>sys_brd_volt_norm : enabled # 電圧の最大値・最小値・不明値が通常値に戻りました。 sys_brd_volt_max : enabled # システムボードの電圧が最大値を超過しました。 sys_brd_volt_min : enabled # システムボードの電圧が最小値を下回りました。 sys_brd_volt_bad : enabled # システムボードの電圧を読み取ることができません。 sys_brd_volt_change : enabled # システムボードの電圧があらかじめ定められた値だけ変化しました。</pre>
制御ボードの 温度イベント	<pre>cb_temp_norm : enabled # システムボードの温度についての説明を参照。 cb_temp_warn : enabled cb_temp_max : enabled cb_temp_911 : enabled cb_temp_bad : enabled cb_temp_change : enabled</pre>
制御ボードの 電圧イベント	<pre>cb_volt_norm : enabled # システムボードの電圧についての説明を参照。 cb_volt_max : enabled cb_volt_min : enabled cb_volt_bad : enabled cb_volt_change : enabled</pre>
センタープレーンの 温度イベント	<pre>centerplane_temp_norm : enabled # システムボードの温度についての説明を参照。 centerplane_temp_warn : enabled centerplane_temp_max : enabled centerplane_temp_911 : enabled centerplane_temp_bad : enabled centerplane_temp_change : enabled</pre>
センタープレーンの 電圧イベント	<pre>centerplane_volt_norm : enabled # システムボードの電圧についての説明を参照。 centerplane_volt_max : enabled centerplane_volt_min : enabled centerplane_volt_bad : enabled centerplane_volt_change : enabled</pre>
センタープレーン サポートボードの 温度イベント	<pre>supp_brd_temp_norm : enabled # システムボードの温度についての説明を参照。 supp_brd_temp_warn : enabled supp_brd_temp_max : enabled supp_brd_temp_911 : enabled supp_brd_temp_bad : enabled supp_brd_temp_change : enabled</pre>
センタープレーン サポートボードの 電圧イベント	<pre>supp_brd_volt_norm : enabled # システムボードの電圧についての説明を参照。 supp_brd_volt_max : enabled supp_brd_volt_min : enabled supp_brd_volt_bad : enabled supp_brd_volt_change : enabled</pre>

システムボードなどの
イベントのタイプ

各コンポーネントに共通のイベントタイプもあります。以下に、イベントをタイプ別に示します。

temp_norm	ボードの温度が過熱状態から通常温度に戻りました。
temp_bad	温度を読み取ることができません。
temp_change	ボード上の重要なコンポーネントの温度が、あらかじめ定められた値だけ変動しました（下記の「事前に定義された値」を参照）。
volt_norm	ボードの電圧が最大電圧または最小電圧の状態から通常の状態に戻りました。
volt_max	ボードの電圧があらかじめ定められた最大値を超えました（下記の「事前に定義された値」を参照）。
volt_min	ボードの電圧があらかじめ定められた最小値を下回りました（下記の「事前に定義された値」を参照）。
volt_bad	電圧を読み取ることができません。
volt_change:	ボードの電圧があらかじめ定められた分量だけ変動しました（下記の「事前に定義された値」を参照）。

IDN 自動リンクの
サポート

InterDomain Networks 機能は、Sun Enterprise 10000 でのみサポートされる点に注意してください。

```
idn_boot : enabled #cpu のシグニチャーの状態が idn_boot を示します。
idn_halt : enabled #cpu のシグニチャーの状態が idn_halt を示します。
idn_awol : enabled #cpu のシグニチャーの状態が idn_awol を示します。
cluster_arbstop : enabled #idn クラスタアービトレーション停止状態
cluster_recordstop : enabled #idn クラスタレコード停止状態
```

IDN イベントタイプ

idn_boot	IDN をサポートするドメインが起動され、対応する IDN ドライバが読み込まれました。このイベントは、該当するドメインを同一の IDN 内の他のメンバーとリンクする自動 IDN リンクが必要であることを edd(1M) に示します。リンクするには、他のメンバーのドメインが起動している必要があります。
idn_halt	関係するドメインで IDN ドライバの読み込みが解除されました。このイベントは、ドメインの自動 IDN リンクと同期をとるために、idn_boot イベントと組み合わせて使用されます。

idn_awol	IDN 内に応答しないまたは状態が不明 (AWOL) なドメインが存在することをドメインが報告しました。このイベントは、AWOL ドメインのリンクを解除するよう edd(IM) に指示します。
cluster_arbstop	IDN 内のドメインがアービトレーション停止 (arbstop) しました。標準のアービトレーション停止イベントでは、指定されたドメインのボードの状態のみ保存されます。ただし、cluster_arbstop イベントでは、IDN 内のすべてのドメインに含まれるすべてのボードの状態が保存されます。
cluster_recordstop	IDN 内のドメインがレコード停止 (recordstop) しました。標準のレコード停止イベントでは、指定されたドメインのボードの状態のみ保存されます。ただし、cluster_recordstop イベントでは、IDN 内のすべてのドメインに含まれるすべてのボードの状態が保存されます。

ホスト回復イベント

```

arbstop : enabled # 特定のドメインのシステムボードのアービトレーション停止状態
recordstop : enabled # 特定のドメインのシステムボードのレコード停止状態
watchdog : enabled # 特定のドメインのシステムボードのウォッチドック状態
environment_shutdown : enabled
reboot : enabled #cpu シグニチャーの状態が再起動状態を示します。
panic1 : enabled #cpu シグニチャーの状態がパニック 1 の状態を示します。
panic2 : enabled #cpu シグニチャーの状態がパニック 2 の状態を示します。
panic_reboot : enabled #cpu シグニチャーの状態がパニック再起動状態を示します。
obp_reset : enabled #cpu シグニチャーの状態が OBP リセット状態を示します。
obpbooting : enabled #cpu シグニチャーの状態が OBP 起動状態を示します。
heartbeat_failure : enabled
#cpu ハートビートビットがハートビート障害を示します。
    
```

ホスト回復イベントの
タイプ

arbstop	特定のドメインに属するシステムボードにアービトレーション停止の状態が発生しました。
recordstop	特定のドメインに属するシステムボードにレコード停止の状態が発生しました。
watchdog	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、ウォッチドッグの状態が発生しました。
environment_shutdown	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、環境シャットダウンの状態が発生しました。
reboot	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、再起動の状態が発生しました。

panic1	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、パニック 1 の状態が発生しました。
panic2	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、パニック 2 の状態が発生しました。
panic_reboot	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、パニック再起動の状態が発生しました。
obp_reset	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) に、OBP リセットの状態が発生しました。
obpbooting	特定のドメインに属するプロセッサ (あるいはプロセッサのセット) が、OBP 起動の状態になりました。この状態はドメインの起動中に発生します。
heartbeat_failure	特定のドメインに属するすべてのプロセッサに、ハートビート障害が発生しました。
その他のイベント	<pre>signature_change : enabled #cpu シグニチャーの状態が変更されました。 system_config_change : enabled # マシンモジュールが追加・削除されました。 sys_brd_power_on : enabled # システムボードの電源が投入されました。 sys_brd_power_off : enabled # システムボードの電源が切断されました。 supp_brd_power_on : enabled # サポートボードの電源が投入されました。 supp_brd_power_off : enabled # サポートボードの電源が切断されました。 cb_power_on : enabled # 制御ボードの電源が投入されました。 cb_power_off : enabled # 制御ボードの電源が切断されました。 bulk_power_norm : enabled # バルク電源が切断状態または障害状態から投入状態へ切り替わりました。 bulk_power_fail : enabled # バルク電源が投入状態から切断状態または障害状態へ切り替わりました。 fan_norm : enabled # ファンが障害状態から動作状態または非動作状態へ切り替わりました。 fan_fail : enabled # ファンが動作状態または非動作状態から障害状態へ切り替わりました。</pre>
その他のイベントのタイプ	
signature_change	プロセッサのシグニチャブロックの状態が変更されました。
system_config_change	システムボード、センタープレーンサポートボード、制御ボード、ファントレイ、バルク電源装置のいずれか 1 つ、または複数の装置がシステムに取り付けられたか、取り外されました。
sys_brd_power_on	システムボードの電源が切断状態から投入状態に切り替わりました。
sys_brd_power_off	システムボードの電源が投入状態から切断状態に切り替わりました。

supp_brd_power_on	センタープレーンサポートボードの電源が切断状態から投入状態に切り替わりました。
supp_brd_power_off	センタープレーンサポートボードの電源が投入状態から切断状態に切り替わりました。
cb_power_on	制御ボードの電源が切断状態から投入状態に切り替わりました。
cb_power_off	制御ボードの電源が投入状態から切断状態に切り替わりました。
bulk_power_norm	バルク電源が切断状態または障害状態から投入状態に切り替わりました。
bulk_power_fail	バルク電源が投入状態から切断状態または障害状態に切り替わりました。
fan_norm:	ファンが障害状態から動作状態または非動作状態に切り替わりました。
fan_fail:	ファンが動作状態または非動作状態から障害状態に切り替わりました。

事前に定義された値 過熱を示す温度限界値、温度変動量、電圧変動量はあらかじめ定義されており、SSP の固定記憶領域に格納されています。ssp_resource(4) を参照してください。

使用例

例 1

```
#
# Event Monitor Configuration File
#
centerplane_temp_warn : enabled
centerplane_volt_max : enabled
```

この例では、システムに対しセンタープレーンの 2 つのイベントタイプの監視を有効化するよう設定しています。

ファイル

以下のファイルがサポートされます。

```
$SSPVAR/etc/platform_name/edd.emc
    インスタンス化された EMC ファイルへのパス
```

関連項目

edd(1M), ssp_config(1M), edd.erc(4)

名前	edd.erc - イベント応答設定ファイル
機能説明	<p>edd.erc ファイルは ASCII ファイルです。このファイルによって、システムが特定のイベントにどのように応答するかが指定されます。イベント応答ファイル (ERC) は、各ドメインに対してそれぞれ 1 つずつと、広域的な (システム全体にわたる) イベント用のものが 1 つあります。各ドメインの ERC ファイルは、そのドメインのシステムボードに固有のイベントだけを処理します。グローバル ERC ファイルは、非システムボード、起動されたドメインに属さないシステムボード、その他ドメインに依存しないコンポーネントのイベントを処理します。</p> <p>グローバル ERC ファイルは、<code>ssp_config(1M)</code> コマンドの実行時にテンプレートファイルに基づいて作成されます。ドメイン ERC ファイルは、<code>domain_create(1M)</code> コマンドの実行時にテンプレートファイルに基づいて作成されます。これらのテンプレートファイルは、<code>\$\$SSPVAR/.ssp_private/templates/Ultra-Enterprise-10000</code> に格納されています。グローバル ERC テンプレートは <code>edd.platform.erc</code>、ドメイン ERC テンプレートは <code>edd.domain.erc</code> という名前になっています。</p> <p>ERC ファイルの各行は、それぞれ次のような形式になっています。</p> <pre>event_type : invoke_action : throttle_timeout : throttle_counter : select_action</pre> <p>上記で</p> <p><i>event_type</i></p> <p>イベントの種類を示すニモニック (名前の文字列) です。下記の「使用例」を参照してください。</p> <p><i>invoke_action</i></p> <p>有効か無効かを示すキーワードです。有効 (<code>enabled</code>) にすると、イベントタイプに対応する応答動作スクリプトが起動されます。無効 (<code>disabled</code>) にすると、応答動作スクリプトは起動されません。このフィールドが空白の場合は、スクリプトは起動されません。</p> <p><i>throttle_timeout</i></p> <p><i>throttle_counter select_action(s)</i> を実行する間隔を秒単位で指定します。<i>throttle_timeout</i> が経過するまで、他の <i>throttle_counter select_action(s)</i> は実行されません。</p> <p><i>throttle_counter</i></p> <p><i>throttle_timeout</i> の時間内に実行可能な <i>select_action</i> の回数を指定します。<i>throttle_timeout</i> が経過すると、<i>throttle_counter select_action(s)</i> の実行が再度可能になります。</p>

select_action

起動する応答動作スクリプトを指定します。このフィールドには、(必要に応じてパス名を付けて)スクリプト名を指定します。パスには、`$$SSPOPT`、`$$SSPVAR`、`$$SUNW_HOSTNAME`などの環境変数を含めてもかまいません。スクリプト名にはオプションの引数を付けることもできます。引数はリテラル(-vなど)またはイベント情報(イベントタイプを示す%e、ボードを示す%b、ボードタイプを示す%t、SNMPトラップデータを示す%d)です。有効なボードタイプ(%t)は次のとおりです。

- 0: タイプなし
- 1: システムボード
- 2: 制御ボード
- 3: センタープレーン
- 4: センタープレーンサポートボード

下記の「使用例」を参照してください。

各フィールドはコロンによって区切られます。コロンの左右にはスペースを入れても入れなくてもかまいません。ハッシュ記号(#)で始まる行はコメントであり、解析されません。

throttle_timeout フィールドと *throttle_counter* フィールドは、処理の実行頻度を制御します。たとえば、*throttle_timeout* を 600 秒、*throttle_counter* を 3 に指定した場合、処理 (*select_action*) は 10 分 (600 秒) ごとに 3 回しか実行されません。処理をこのように制限することによって、ログメッセージやダンプファイルの繰り返しを減らすことができます。こういった制限はシステムボードやドメインごとに個別に適用できます。たとえば、特定のボードに対する *sys_brd_temp_max* イベント処理が別のボードの *sys_brd_temp_max* イベント処理を制限することはありません。同様に、特定のドメインの *arbstop* イベント処理が別のドメインの *arbstop* イベント処理を制限することはありません。

ERC ファイルでは、特定のイベントについて複数の応答動作を定義することができます。2 つ目の応答動作スクリプトを指定するには、イベントタイプ名を 1 行目と同じにして 2 行目に記述します。応答動作スクリプトは、(並行してではなく) イベント応答設定ファイルに記述されている順に 1 つずつ実行されます。1 つのイベントについて複数の応答動作スクリプトが存在する場合は、引数の %p と %s を使用して、直前の応答動作スクリプトの名前と終了ステータスを現在の応答動作スクリプトに渡すことができます。

edd.platform.erc ファイルと *edd.domain.erc* ファイルには、下記の情報が格納されています。イベントタイプの詳細は、*edd.emc(4)* を参照してください。ERC ファイルの内容は、次のように設定されています。

システムボードの
イベント

システムボードのイベント
 制御ボードのイベント
 センタープレーンのイベント
 センタープレーンサポートボードのイベント
 IDN のイベント
 CBS/CBE 接続のイベント
 システム設定変更のイベント
 ホスト回復のイベント
 その他のイベント

```
sys_brd_temp_norm : enabled : 0 : 1 : TempNormact -b %b -e %e -d %d -t %t
sys_brd_temp_high : enabled : 300 : 1 : TempHighact -b %b -e %e -d %d -t %t
sys_brd_temp_warn : enabled : 300 : 1 : TempWarnact -b %b -e %e -d %d -t %t
sys_brd_temp_max : enabled : 300 : 1 : TempMaxact -b %b -e %e -d %d -t %t
sys_brd_temp_911 : enabled : 60 : 1 : Temp911act -b %b -e %e -d %d -t %t
sys_brd_temp_bad : enabled : 300 : 1 : TempBadact -b %b -e %e -d %d -t %t
sys_brd_volt_norm : enabled : 0 : 1 : VoltageNormalact -b %b -e %e -d %d -t %t
sys_brd_volt_max : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
sys_brd_volt_min : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
sys_brd_volt_bad : enabled : 300 : 1 : VoltageBadact -b %b -e %e -d %d -t %t
```

システムボードの
応答動作

sys_brd_temp_norm
 ボードの温度が過熱状態から通常の状態に戻ったことを知らせるメッセージをログに書き込みます。

sys_brd_temp_high または **sys_brd_temp_warn**
 ボードの温度が高過ぎることを知らせるメッセージをログに書き込みます。

sys_brd_temp_max
 必要に応じて以下のステップを実行し、温度上限値超過イベントを処理します。

1. そのボードが IDN に属する場合は、そのドメインのリンクを解除します。
2. そのボードがいずれかのドメイン内にある場合は、ドメインをシャットダウンします。
3. ボードの電源を切断します。
4. そのボードがいずれかのドメイン内にあり、かつ同じドメイン内で他のボードの電源が投入されていれば、ドメインを再起動します。

sys_brd_temp_911
 そのボードが IDN に属する場合は、そのドメインのリンクを解除します。
 ドメインに属しているかどうかに関わらず、ボードの電源を切断します。

```

sys_brd_temp_bad
    ボードの温度を読み取ることができなかったことを知らせるメッセージをログに書き込みます。
sys_brd_volt_norm
    ボードの電圧が通常の状態に戻ったことを知らせるメッセージをログに書き込みます。
sys_brd_volt_max
    ボードの電圧が上限値を超えたことを知らせるメッセージをログに書き込みます。
sys_brd_volt_min
    ボードの電圧が下限値を下回ったことを知らせるメッセージをログに書き込みます。
sys_brd_volt_bad
    ボードの電圧を読み取ることができなかったことを知らせるメッセージをログに書き込みます。

```

制御ボードのイベント

```

cb_temp_norm : enabled : 0 : 1 : TempNormact -b %b -e %e -d %d -t %t
cb_temp_high : enabled : 300 : 1 : TempHighact -b %b -e %e -d %d -t %t
cb_temp_warn : enabled : 300 : 1 : TempWarnact -b %b -e %e -d %d -t %t
cb_temp_max : enabled : 300 : 1 : TempMaxact -b %b -e %e -d %d -t %t
cb_temp_911 : enabled : 60 : 1 : Temp911act -b %b -e %e -d %d -t %t
cb_temp_bad : enabled : 300 : 1 : TempBadact -b %b -e %e -d %d -t %t
cb_volt_norm : enabled : 0 : 1 : VoltageNormalact -b %b -e %e -d %d -t %t
cb_volt_max : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
cb_volt_min : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
cb_volt_bad : enabled : 300 : 1 : VoltageBadact -b %b -e %e -d %d -t %t

```

制御ボードの応答動作

```

cb_temp_norm
    ボードの温度が過熱状態から通常の状態に戻ったことを知らせるメッセージをログに書き込みます。
cb_temp_high または cb_temp_warn
    ボードの温度が高過ぎることを知らせるメッセージをログに書き込みます。
cb_temp_max
    システムに設定されている制御ボードが 1 つだけの場合は、システム内のすべてのドメインをシャットダウンし、すべての電源を切断します。2 つの制御ボードが設定されている場合は、最大温度を超えている制御ボードの電源を切断します。
cb_temp_911
    システム全体をシャットダウンします。
cb_temp_bad
    ボードの温度を読み取ることができなかったことを知らせるメッセージをログに書き込みます。
cb_volt_norm
    ボードの電圧が通常の状態に戻ったことを知らせるメッセージをログに書き込みます。

```

センタープレーンの
イベント

cb_volt_max
ボードの電圧が上限値を超えたことを知らせるメッセージをログに書き込みます。

cb_volt_min
ボードの電圧が下限値を下回ったことを知らせるメッセージをログに書き込みます。

cb_volt_bad
ボードの電圧を読み取ることができなかったことを知らせるメッセージをログに書き込みます。

centerplane_temp_norm : enabled : 0 : 1 : TempNormact -b %b -e %e -d %d -t %t
centerplane_temp_high : enabled : 300 : 1 : TempHighact -b %b -e %e -d %d -t %t
centerplane_temp_warn : enabled : 300 : 1 : TempWarnact -b %b -e %e -d %d -t %t
centerplane_temp_max : enabled : 300 : 1 : TempMaxact -b %b -e %e -d %d -t %t
centerplane_temp_911 : enabled : 60 : 1 : Temp911act -b %b -e %e -d %d -t %t
centerplane_temp_bad : enabled : 300 : 1 : TempBadact -b %b -e %e -d %d -t %t
centerplane_volt_norm : enabled : 0 : 1 : VoltageNormalact -b %b -e %e -d %d -t %t
centerplane_volt_max : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
centerplane_volt_min : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
centerplane_volt_bad : enabled : 300 : 1 : VoltageBadact -b %b -e %e -d %d -t %t

センタープレーンの
応答動作

centerplane_temp_norm
センタープレーンの温度が過熱状態から通常の状態に戻ったことを知らせるメッセージをログに書き込みます。

centerplane_temp_high または **centerplane_temp_warn**
センタープレーンの温度が高過ぎることを知らせるメッセージをログに書き込みます。

centerplane_temp_max
残りのドメインをすべてシャットダウンし、システムの電源を切断します。

centerplane_temp_911
システム全体をシャットダウンします。

centerplane_temp_bad
センタープレーンの温度を読み取ることができなかったことを知らせるメッセージをログに書き込みます。

centerplane_volt_norm
センタープレーンの電圧が通常の状態に戻ったことを知らせるメッセージをログに書き込みます。

centerplane_volt_max
センタープレーンの電圧が上限値を超えたことを知らせるメッセージをログに書き込みます。

centerplane_volt_min
センタープレーンの電圧が下限値を下回ったことを知らせるメッセージをログに書き込みます。

centerplane_volt_bad
センタープレーンの電圧を読み取ることができなかったことを知らせるメッセージをログに書き込みます。

センタープレーン
サポートボードの
イベント

```

supp_brd_temp_norm : enabled : 0 : 1 : TempNormact -b %b -e %e -d %d -t %t
supp_brd_temp_high : enabled : 300 : 1 : TempHighact -b %b -e %e -d %d -t %t
supp_brd_temp_warn : enabled : 300 : 1 : TempWarnact -b %b -e %e -d %d -t %t
supp_brd_temp_max : enabled : 300 : 1 : TempMaxact -b %b -e %e -d %d -t %t
supp_brd_temp_911 : enabled : 60 : 1 : Temp911act -b %b -e %e -d %d -t %t
supp_brd_temp_bad : enabled : 300 : 1 : TempBadact -b %b -e %e -d %d -t %t
supp_brd_volt_norm : enabled : 0 : 1 : VoltageNormalact -b %b -e %e -d %d -t %t
supp_brd_volt_max : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
supp_brd_volt_min : enabled : 300 : 1 : Voltageact -b %b -e %e -d %d -t %t
supp_brd_volt_bad : enabled : 300 : 1 : VoltageBadact -b %b -e %e -d %d -t %t
    
```

センタープレーン
サポートボードの
応答動作

```

supp_brd_temp_norm
    ボードの温度が過熱状態から通常の状態に戻ったことを知らせる
    メッセージをログに書き込みます。
supp_brd_temp_high または supp_brd_temp_warn
    ボードの温度が高過ぎることを知らせるメッセージをログに書き
    込みます。
supp_brd_temp_max
    実行中のドメインをすべてシャットダウンし、システムの電源を
    切断します。
supp_brd_temp_911
    システムをシャットダウンします。
supp_brd_temp_bad
    センタープレーンサポートボードの温度を読み取ることができな
    かったことを知らせるメッセージをログに書き込みます。
supp_brd_volt_norm
    センタープレーンサポートボードの電圧が通常の状態に戻ったこ
    とを知らせるメッセージをログに書き込みます。
supp_brd_volt_max
    センタープレーンサポートボードの電圧が上限値を超えたことを
    知らせるメッセージをログに書き込みます。
supp_brd_volt_min
    センタープレーンサポートボードの電圧が下限値を下回ったこと
    を知らせるメッセージをログに書き込みます。
supp_brd_volt_bad
    センタープレーンサポートボードの電圧を読み取ることができな
    かったことを知らせるメッセージをログに書き込みます。
    
```

IDN イベント

```

idn_boot : enabled : 20 : 1 : IDNevent -e %e -d %d
idn_halt : enabled : 20 : 1 : IDNevent -e %e -d %d
idn_awol : enabled : 30 : 1 : IDNevent -e %e -d %d
cluster_arbstop : enabled : 1800 : 3 : Arbstopact -d %d
cluster_recordstop : enabled : 1800 : 3 : Recordstopact -d %d
    
```

IDN イベントの 応答動作

idn_boot
 同じ IDN 内にある他のドメインの内、少なくとも 1 つのドメインが起動しており、しかも IDN ソフトウェアを読み込んでいる場合は、**domain_link(1M)** コマンドを実行して該当するドメインを IDN にリンクします。

idn_halt
 IDN の内部状態についての情報を更新して、すべての IDN のメンバードメインの正確な状態を維持するための IDN イベント処理ルーチンを有効にします。

idn_awol
 AWOL ドメインが存在することをローカルドメインが報告します。AWOL ドメインが停止している場合には、そのドメインのリンクを関係するドメインから解除することにより、残りの IDN メンバードメインが通信を再開できるようにします。

cluster_arbstop
 すべての IDN メンバードメインの **hpost(1M)** ダンプを実行し、ブートプロセッサの **BBRAM** 情報を保存します。それからすべての IDN メンバードメインの完全な立ち上げを実行します。

cluster_recordstop
 すべての IDN メンバードメインの **hpost(1M)** ダンプを実行し、次にすべての IDN メンバードメインのレコード停止のクリアを試行します。

CBS/CBE 接続の イベント

```
cbe_connected : enabled : 0 : 1 : actionsysclock
cbe_connected : enabled : 0 : 1 : actioncb
cbe_connected : enabled : 0 : 1 : PowerFailRebootact
```

CBE/CBS 接続の 応答動作

必要に応じてシステムクロックとファンの回転速度を設定し、制御ボードのハートビートを再確立します。以前に起動されていたすべてのドメインのオペレーティングシステムと電源の状態をチェックし、電源障害が原因となってドメインを再起動する必要があるかどうかを判別します。このイベントは、**SSP** の **CBS** デーモンと制御ボードエグゼクティブ (**CBE**) との接続が失われた状態を表しています。

ホスト回復イベント

```
arbstop : enabled : 900 : 3 : Arbstopact -d %d
recordstop : enabled : 900 : 3 : Recordstopact -d %d
watchdog : enabled : 900 : 3 : WatchDogRebootact -d %d
environment_shutdown : enabled : 600 : 1 : Environmentact -d %d
obp_reset : enabled : 300 : 3 : ObpResetact -d %d
cb_power_on : enabled : 0 : 1 : PowerOnact -t %t -b %b
cb_power_off : enabled : 0 : 1 : PowerOffact -t %t -b %b
reboot : enabled : 300 : 3 : Rebootact -d %d
panic1 : enabled : 900 : 3 : Panicact -t 300 -d %d -e %e
panic2 : enabled : 900 : 3 : Panicact -t 900 -d %d -e %e
panic_reboot : enabled : 900 : 3 : PanicRebootact -d %d
heartbeat_failure : enabled : 900 : 3 : HeartBeatFailact -d %d
```

ホスト回復の応答動作

arbstop	ドメインに対して hpost(1M) ダンプを実行し、完全な立ち上げを実行します。
recordstop	hpost(1M) ダンプを実行し、レコード停止のクリアを試みます。
watchdog	以下のステップを実行します。 <ol style="list-style-type: none">1. ドメイン内のすべてのプロセッサのリセット情報 (resetinfo) をダンプします。2. ドメイン内のすべてのプロセッサのシグニチャブロックをダンプします。3. ドメインに対して hpost(1M) ダンプを実行します。4. 完全な立ち上げを実行して、ドメインを再起動します。
environmental_shutdown	特定のドメインに環境シャットダウンの状態が検出されたことを知らせるメッセージをログに書き込みます。
obp_reset	OBP リセットが発生したことを知らせるメッセージをログに書き込みます。
cb_power_on	制御ボードの電源が投入されたことを知らせるメッセージをログに書き込みます。
cb_power_off	制御ボードの電源が切断されたことを知らせるメッセージをログに書き込みます。
reboot	高速立ち上げを実行することによって、ユーザーが要求した再起動を実行します。
panic1	Panicact フィールドの -t オプションで指定された時間だけスリープし、その後ドメインが依然としてパニック 1 の状態である場合には、高速立ち上げを実行します。
panic2	Panicact フィールドの -t オプションで指定された時間だけスリープし、その後ドメインが依然としてパニック 2 の状態である場合には、高速立ち上げを実行します。
panic_reboot	完全な bringup を実行することによってシステムを再起動します。
heartbeat_failure	完全な bringup を実行することによってシステムを再起動します。

その他のイベント

```

sys_brd_power_on : enabled : 0 : 1 : PowerOnact -t %t -b %b
sys_brd_power_off : enabled : 0 : 1 : PowerOffact -t %t -b %b
supp_brd_power_on : enabled : 0 : 1 : PowerOnact -t %t -b %b
supp_brd_power_off : enabled : 0 : 1 : PowerOffact -t %t -b %b
bulk_power_norm : enabled : 0 : 1 : BulkPowerNormact -d %d
bulk_power_fail : disabled : 300 : 1 : BulkPowerFailact -d %d
fan_norm : enabled : 0 : 1 : FanNormact -d %d
fan_fail : enabled : 180 : 1 : FanFailact -d %d
system_config_change : enabled : 0 : 1 : SystemConfChangeact -d %d

```

その他のイベントの
応答動作

```

sys_brd_power_on および sys_brd_power_off
システムボードの電源が投入状態または切断状態になったことを
知らせるメッセージをログに書き込みます。
supp_brd_power_on および supp_brd_power_off
センタープレーンサポートボードの電源が投入状態または切断状
態になったことを知らせるメッセージをログに書き込みます。
bulk_power_norm
48 ボルト電源が入っていることを知らせるメッセージをログに書
き込みます。(メッセージによっては「48 volt power」の代わりに
「bulk power」という表現が使われることがあります)。
bulk_power_fail
48 ボルト電源が障害状態または切断状態であることを知らせる
メッセージをログに書き込み、現在有効な電源だけでシステムの
稼働を継続できるかどうかを判別します。継続できない場合に
は、システム全体の電源を切断します。
fan_norm
ファンが障害状態から動作状態または非動作状態になったことを
知らせるメッセージをログに書き込みます。
fan_fail
ファンが動作状態または非動作状態から障害状態になったことを
知らせるメッセージをログに書き込みます。
system_config_change
システムボード、センタープレーンサポートボード、制御ボー
ド、ファントレー、48 ボルト電源のすべて、あるいはいずれかが
システムに取り付け・取り外されたことを知らせるメッセージを
ログに書き込みます。

```

使用例

例 1 グローバル ERC ファイル

```

# Event Response Configuration File
#
centerplane_temp_warn : enabled : 300 : 1 : TempWarnact -b %b -e %e -d %d -t %t
centerplane_temp_warn : enabled : 300 : 1 : fans -hi
centerplane_temp_norm : enabled : 0 : 1 : fans -off

```

これはグローバル ERC ファイルの例です。最初の 2 行は、センタープレーンの温度超過のイベントに対する応答方法を定義しています。1 行目は、指定した情報 (イベントが発生しているボードの番号 (%b)、イベントタイプ (%e)、SNMP トラップデータ (%d)、ボードタイプ (%t)) を

TempWarnact という動作スクリプトに渡してから、TempWarnact を実行することを示しています。2 行目は、高速回転の設定でファンの電源を投入することを示しています。

3 行目は、センタープレーンの温度が通常に戻ったときに、ファンの電源を切断することを示しています。

ファイル

以下のファイルがサポートされます。

`$SSPVAR/etc/platform_name/edd.erc`

インスタンス化されたグローバル ERC ファイルへのパス

`$SSPVAR/etc/platform_name/domain_name/edd.erc`

インスタンス化されたドメイン ERC ファイルへのパス

関連項目

edd(1M), edd.emc(4)

名前	fad_files - ファイルアクセスデーモンファイル
注意	このファイルは手動では編集しないでください。このファイルはドメイン管理用のツールとコマンドによって自動的に保守されます。
機能説明	<p>fad(1M) は fad_files ファイルを調べて、監視すべきファイルアクセスデーモンファイルを判別し、これらのファイルの格納場所と割り当てられているアクセス権を判別します。fad(1M) はさらに、fad_files の情報を使用して、SSP ファイルの読み取り、書き込み、および変更通知の同期をとります。ネットワーク上の他の SSP デーモンやクライアントもこのファイルにアクセスして、ファイルアクセスデーモンの情報を取得します。</p> <p>fad_files ファイルは、\$SSPVAR/.ssp_private に格納されています。通常、次の形式の行が 1 行または複数行含まれています。</p> <pre>type permission path sub_path filename</pre> <p>上記で、</p> <p><i>type</i> ファイルタイプ (domain または nondomain) を示します。</p> <p><i>permission</i> アクセス権を示します。アクセス権は、r (読み取り専用)、w (書き込み専用)、または rw (読み書き) のいずれかです。fad(1M) は適切なアクセス権が割り当てられているファイルに対してのみ、読み取り・書き込みを行うことができます。</p> <p><i>path</i> Sun Enterprise 10000 の有効なパスです。<i>path</i> には SSPOPT、SSPVAR、SSPETC などの任意の SSP 環境変数を指定できます。</p> <p><i>sub_path</i> Sun Enterprise 10000 システム下での絶対パスです。<i>sub_path</i> は通常、etc、data、または . (ドット) です。</p> <p><i>filename</i> fad(1M) の実際のファイル名です。</p> <p>ハッシュ記号 (#) で始まる行はコメントとみなされます。</p>

使用例

例 1 fad(4) ファイルの例

```
#
# SSP-Enterprise 10000: fad_files example
#
nondomain rw SSPVAR .ssp_private domain_config
domain rw SSPVAR etc eeprom.image
nondomain r SSPETC etc ssp_startup.main
```

注意事項

このファイルは SSP パッケージに付属しており、SSP パッケージと一緒にインストールされます。

関連項目

fad(1M)

名前	postrc - (.postrc) hpost 属性ファイル
注意	.postrc 命令の多くは設計・製造の段階でのみ使用します。現場で使用できる命令についても、購入先の技術者またはシステム管理者だけが使用してください。これらの機能について十分に理解していないユーザーは、このファイルを使用しないでください。
機能説明	<p>.postrc ファイルは ASCII テキストファイルであり、hpost(1M) プログラム用の各種の設定属性を定義しています。POST のオプションは、このファイルによって制御されます。</p> <p>hpost(1M) の機能のいくつかは、コマンド行からも制御することができます。コマンド行から制御できる機能については、以下の説明の中で使用法を明示してあります。コマンド行の引数は .postrc ファイル内の指定よりも優先されます。 .postrc ファイル内の指定はデフォルト設定よりも優先されます。 hpost --?postrc と入力すると、.postrc のオプションと構文に関する簡単な説明が表示されます。</p> <p>最初の引数が -n でない場合は、hpost(1M) はオプションファイルの .postrc を読み取り、そのファイルに記述されている命令を実行してから、ホストとのやり取りを開始します。 .postrc が見つからない場合は、hpost(1M) は .postrc なしで処理を続行します。hpost(1M) は、まず現在のディレクトリ (.) の中で .postrc を探します。ここで見つからない場合は、続いて \$SSPVAR/etc/platform_name/\$SUNW_HOSTNAME を探します。ここでも .postrc が見つからない場合は、ユーザーのホームディレクトリである \$HOME を探します。ただし例外として、カレントディレクトリが \$HOME の場合は、検索パスの最初の要素である (.) は省略されます。この例外があると、ユーザーのホームディレクトリから実行するときにはドメイン固有の .postrc ファイルがユーザーのデフォルトの \$HOME/.postrc よりも優先されることになり、ユーザーがホームディレクトリ以外の任意のディレクトリにいるときには、ローカルの ./postrc が優先されることになります。</p> <p>以下に説明するコマンドのいくつかは、ローカルファイル ./postrc が使用された場合にのみ有効です。こうしたコマンドについてはその旨を明示します。カレントディレクトリが \$HOME である場合は、 ./postrc 検索パスは省略されるため、この種のコマンドは使用できません。</p> <p>.postrc ファイルには、以下の規則が適用されます。</p> <ul style="list-style-type: none"> ■ キーワードでは大文字と小文字が区別されます。 ■ 数値オプションの引数は通常は 10 進数です。ただし、先頭に x または 0x を付けることによって、16 進数で指定することもできます。 ■ キーワードとオプションの引数は空白で区切られます。 ■ ハッシュ記号 (#) で始まる行部分はコメントとみなされ、無視されません。

属性

以下に、属性をアルファベット順で示します。属性はユーザーレベルによって分類できます。ユーザーレベルは、属性名の末尾の括弧内の番号によって示します。各レベルの意味は、以下のとおりです。

- Level 1 システム管理者が使用するコマンドです。これらのコマンドは、システム管理者用の機能、または(たとえば logfile のように)無害で安全な機能を制御します。
- Level 2 サンおよび購入先の技術者だけが使用するコマンドです。
- Level 3 サンの POST 開発者が(デバッグ用に)使用するコマンドであり、サンの技術者だけが使用します。

allow_lab-only_components (Level 3)

サンでの試験の際には、市販のマシンでは使用しないようなコンポーネントを組み込んだシステムを動作させることがあります。たとえば、既知のバグや速度上の問題を抱えたままのプロトタイプのコポーネントを組み込んだシステムを試すような場合です。問題のコポーネントがプロダクションシステムに害を与えることはありません。通常、POST は Component ID (CID) 値に基づいて、こうしたコンポーネントを即座に拒否します。このため、試験の目的を果たすには、allow_lab-only_components コマンドを使用して、POST に対し、CID ベースの拒否を実行せずに、特定の値を実験専用として認識するように指示します。このコマンドは、使用可能な CID のリストを拡張するだけであり、すべての CID チェックを無効にするわけではありません。

allow_sys_errctl (Level 3)

PC、CIC、または MC ASIC を JTAG 設定する場合に、エラー状態、エラーマスク、強制エラーのレジスタへの書き込み・クリアのアクセスを可能にします。通常、このコマンドを有効にするのは、診断コード (POST など) が実行されており、まだ最終設定の状態になっていないときだけです。

alt_level level (Level 1)

-a コマンド引数が呼び出された場合は、デフォルトの代替レベルではなく、level を POST レベルとして使用します。ただし、そのレベルより高い値が指定された level コマンドが呼び出された場合には、その値が使用されます。

この -a メカニズムは、特定のエラー再起動シナリオで通常より高いレベルの診断を実行するためのものです。ただし、alt_level で指定されたレベルにかかわらず、-a を呼び出さないときよりも低いレベルで hpost が実行されることはありません。

arb_flip_ctr_limit *value* (Level 3)

Sun Enterprise 10000 のすべてのアービトレーション ASIC が、アービトレーションアルゴリズムの公平度を改善するための許可をいくつか実行した後で、クライアント要求の優先順位を逆転します。ASIC を設定する際に許可の数をプログラムすることができます。通常はデフォルト値が使用されます。サンの技術者は、このコマンドを使用して許可の数の試験を行います。

asic_ignore *asic_type* (Level 3)

ASIC の種類のテスト、設定、ダンプを実行しないコマンドです。サンの技術者は、このコマンドを使用して、特定の ASIC の種類が欠如している開発システムで POST コードをテストします。このような場合に、このコマンドを利用しないと、POST は直ちに失敗します。host -?postrc を入力すると、*asic_type* キーワードのリストが表示されます。

assume_ldarb_rev_2 (Level 3)

LDARB ASIC の CID に対し、設定方法を調べるためのチェックをせずに、rev 1 ではないとみなします。このコマンドは、特別なモードの POST (scantool_simulate を参照) をサポートします。この特別なモードは、ASIC 入力が不可能な Verilog ASIC シミュレーションへの入力として、JTAG 設定を提供するときに使用されます。

blacklist_file {*file* | none} (Level 2)

コマンド行での同等機能: -X

デフォルト (\$SSPVAR/etc/*platform_name*/blacklist.*file*) の代わりに、指定された blacklist(4) ファイルを使用します。*file* に blacklist(4) ファイルへのパスを指定するか、none を指定します。none を指定した場合は、POST は blacklist ファイルを読み込みません。

プロダクションシステムで blacklist_file を使用する場合には注意が必要です。POST が標準外の blacklist(4) ファイルを使用していることは、他の SSP ソフトウェアにはわかりません。

board_red_any_fail (Level 2)

ボード上で障害が検出されたときに、そのボードのすべてのリソースを即座にレッドリストに入れます。このコマンドの目的は、ボードの設定が続行されるのを防ぎ、障害の事後分析を可能にすることです。sytem_red_any_fail および redlist(4) も参照してください。

bus_shuffle_mode {0|1|2} (Level 3)

PC ASIC の機能によって、GAB へのアドレスの割り当てをデフォルトから変更することができます。デフォルトの割り当ては PA[8:6] に基づきます。割り当ての詳細は GAB の数と種類に依存します。このコマンドのデフォルトはシャフルモード 0 です。シャフルモード 1 は PA[8:6] XOR PA[18:16] に、シャフルモード 2 は PA[8:6] XOR PA[20:18] に基づいています。サンの技術者は、このコマンドを使用して、シャフル機能のさまざまな効果を試すことができます。

cplane_initial_config (Level 3)

コマンド行での同等機能 : -c

POST を (後続のドメインではなく) 最初のドメインで開始します。Sun Enterprise 10000 システムでは、複数の独立したドメインを実行することができます。ただし、それらのドメインが isolated_sysboard モードにない限り、それらのドメインはすべてセンタープレーン ASIC を共有します。センタープレーン ASIC は単一の機能バス設定に含まれます。デフォルトでは、POST は、現在立ち上げ中の最初のドメインで実行されていないことを想定します。したがって、POST はセンタープレーンを調べてバス設定を判別し、その設定だけを使用します。POST はセンタープレーン ASIC の機能設定も行いません。通常、POST を最初のドメインで開始するには、コマンド行オプションの -c を使用します。最初のドメインで実行される場合は、POST はバス設定を確立し、**加えて**センタープレーン ASIC の初期設定を行う必要があります。

このコマンド (およびこれと同等の -c コマンド行オプション) は、POST を最初のドメインで開始します。最初のドメインでは、バス設定を確立し、**加えて**センタープレーン ASIC の初期設定を行う必要があります。

cplane_initial_config は、コマンド行オプションの -c に相当します。cplane_initial_config を使用するのには、初期化されていないシステムで POST を頻繁に実行する必要があるサンの技術者や保守サービス担当者です。

複数のドメインを実行している (あるいは実行する可能性がある) プロダクションシステムでは、このコマンドを使用しないでください。2 番目に立ち上げられたドメインで実行される POST が、最初に立ち上げられたドメインで実行されているプログラムに障害を発生させるおそれがあります。

debug_mask mask (Level 2/3)

mask はデバッグマスクとして使用します。このコマンドは、試験的コードや POST の正体不明の動作モードを制御するためのビットマスクであり、主に POST 開発者が使用します。*mask* を内部マスクに OR 処理し、ビットごとに 1 行を設定することもできます。これを行うことによって、.postrc ファイルのリストをコメント化して、ビットを追加または削除できるようになります。これらのビットは時間とともに変化するため、現在のビットを知るには、`hpost --?postrc` を実行する必要があります。

default_ecache_mbytes {H|1|2|4|8|16|32} (Level 2)

プロセッサテストの初期段階で行われる外部キャッシュ (ecache) の検査を省略する場合に、すべてのプロセッサについて、指定した外部キャッシュ (ecache) サイズが想定されます。検査を省略し、ここに指定した値が不正である場合、システムはキャッシュの整合性を維持できなくなります。default のデフォルト ("default default") は、0.5MB - "H" です。

default_memgrp_mbytes {64|256|1024} (Level 2)

指定した DIMM グループのサイズをデフォルトとして使用します。DIMM 検査フェーズに先立つ FOM(Figure of Merit) 計算には、このデフォルトのサイズが使用されます。また、`skip_phase` コマンドで DIMM 検査フェーズを省略する場合には、ここで指定した値が、良好な DIMM グループのサイズとみなされます。default のデフォルト ("default default") は 64MB です。

display_fom_calc {off|on} (Level 1)

このコマンドを on に設定すると、POST は FOM(Figures of Merit) を計算して設定を選択する際に、45 のバス設定のすべてについて計算済みの FOM を出力します。off に設定した場合は、この出力は行われません。このコマンドが指定されていない場合に POST が FOM を出力するのは、冗長 (verbose) レベルが一定の値以上に設定されているときです。

dom_transgress_err_enbl (Level 2)

センタープレーンの設定時に GDARB ASIC が Domain Transgression Errors (ドメイン違反エラー) を報告できるようにします。これは、-C オプションを指定して hpost を起動した場合に実行されます (cplane_initial_config を参照)。違反エラーは、システムボードのデータ決定者が GDARB の SMD マスク構成レジスタの定義とは異なるドメインクラスタのボードにデータを送信するよう要求したときに発生します。デフォルトでは、これらのエラーの報告は無効になっています。これは、ボードの障害によってパリティエラーと違反エラーの両方が発生したときに GDARB がアービトレーション停止を広範囲にわたって行う可能性を最小限に抑えるためです。GDARB は違反要求を禁止しますが、エラーとしては報告しません。追加されたエラー検出方法が追加されたドメイン障害の検出方法よりも望ましい場合には、dom_transgress_err_enbl コマンドを使用してエラーの報告機能を有効にすることができます。

domain boardmask (Level 3)

システムボードのセットをドメインとして宣言します。ドメインは、Sun Enterprise 10000 システム内のシステムボードのセットであり、共有メモリー (shared_mem コマンドを参照) を除き、各セットは相互に独立しています。この .postrc コマンドはプロダクションシステムでのドメインを管理するものではなく、ドメインレジスタを設定するための POST への静的命令であり、サンの技術者がドメインハードウェアのデバッグのために使用します。この domain コマンドが有効になるのは、カレントディレクトリの .postrc ファイルを使用した場合だけです。ローカルディレクトリ以外の .postrc を使用した場合は、POST はこのコマンドを無視し、警告メッセージを表示します。

download_bbsram_verify {off|on} (Level 2)

このコマンドを on に設定すると、プロセッサ BBSRAM への JTAG POST コードのダウンロードが実際に行われたかどうかを検査します。off に設定すると、この検査は行われません。このコマンドを使用しない場合に検査が行われるのは、レベルが一定の値以上に設定されているときです。検査を実行するために、download_bbsram_verify は BBSRAM からコードをアップロードし、コードファイルデータと比較します。JTAG ブロードキャストで行われるダウンロードとは異なり、アップロードと比較は各プロセッサについて順次行う必要があります。

download_path *dirpath* (Level 2)

dirpath 内にある POST 用のダウンロードファイルを使用します。デフォルトは \$SSPOPT/release/Ultra-Enterprise-10000/*/*/*/*/hostobjs です (*/*/*/*/ はリリースレベルの依存状態を示します)。

dtag_test_all_procs (Level 2)

通常、POST はボード上の各 CIC について、各プロセッサに対して自らの DTAG RAM をテストするように割り当てます。これによって、テスト済みのリソースだけを使用できることとなります。欠如しているプロセッサまたは障害が発生しているプロセッサの DTAG RAM は他の用途には使用できません。ただし、ある種の状況では (たとえば製造テストでは)、将来のプロセッサ増加を見込んですべての DTAG RAM をテストします。このコマンドは、DTAG RAM テストのフェーズで、各プロセッサに対し、自身の RAM だけでなく、CIC 上のすべての RAM をテストするように指示します。同様の理由から、現在装着済みプロセッサモジュールの外部キャッシュ (ecache) のサイズをサポートする DTAG SRAM だけではなく、物理的に装着されているすべての DTAG SRAM のテストを実行します。

dump_no_sneeprom (Level 3)

コマンド行オプションの -D を使用してダンプファイルを作成する場合や、POST 実行時のエラーによってダンプファイルが自動的に作成される場合は、ダンプ内にシリアル番号 EEPROM の内容を含めないようにします。このコマンドを使用することによって、一定の環境の下でのダンプ作成を大幅に高速化することができます。このコマンドは将来的には削除される可能性があります。

dump_on_exit [*boardmask* [*path*]] (Level 2)

POST の終了直前に、POST にシステム状態のダンプファイルを作成させます。*boardmask* はダンプされるボードを制御します。ビット [15:0] は、システムボード、[17:16] はセンタープレーンハーフを制御します。ボードがダンプされるのは、対応ビットが 1 の場合です。デフォルトは x3FFFF (すべてをダンプ) です。値は通常、16 進数で指定されます。*path* はダンプファイルのデフォルトのパスを変更します。*path* を指定できるのは、*boardmask* が指定されている場合だけです。どちらのオプションもコマンド行オプションの -D と同様です。

dump_pathname *dirpath* (Level 2)

プログラムのデフォルトディレクトリ (\$SSPVAR/adm/\$SUNW_HOSTNAME) ではなく、指定したデフォルトディレクトリにダンプファイルを作成します。コマンド行オプションの (-D) または `dump_on_exit` コマンドで指定したパスは、`dump_pathname` で指定したパスよりも優先されます。

fom_weightabus *float-in_range* (Level 1)

FOM 計算で、アドレスバスの数の重みを指定します。*float-in_range* の有効値は 1.0 ~ 5.0 であり、デフォルトは 1.0 です。

fom_weightdbus *float-in_range* (Level 1)

FOM 計算で、データバス (72 ビットデータバス) の数の重みを指定します。*float-in_range* の有効値は 1.0 ~ 5.0 (境界値含む) であり、デフォルトは 1.0 です。

fom_weightmem *float-in_range* (Level 1)

FOM 計算で、メモリーの重みを指定します。*float-in_range* の有効値は 1.0 ~ 5.0 (境界値含む) であり、デフォルトは 1.0 です。

fom_weightproc *float-in_range* (Level 1)

FOM 計算で、プロセッサの重みを指定します。*float-in_range* の有効値は 1.0 ~ 5.0 (境界値含む) であり、デフォルトは 1.0 です。

fom_weightscard *float-in_range* (Level 1)

FOM 計算で、SBus カードの重みを指定します。*float-in_range* の有効値は 1.0 ~ 5.0 (境界値含む) であり、デフォルトは 1.0 です。

force_dtag_parity_mode (Level 2)

Sun Enterprise 10000 システムボードにリビジョン 2 CIC ASIC が装着されている場合、通常、POST は CIC を Dtag ECC (Error Checking and Correction) モードに設定します。Dtag ECC モードのサポートには、オンボードのプロセッサに最小限の外部キャッシュ (Ecache) が必要です。ECC は、ドメインの操作に影響を与えないで CIC ハードウェアによって訂正される Dtag SRAM 読み取り内のシングルビットエラーを許可します。しかし、パリティモードにおけるシングルビットエラーでは、致命的なアービトレーション停止エラーが発生します。この属性を使用すると、ECC モードが利用可能な場合でも、POST は常に CIC を Dtag パリティモードに設定するようになります。

force_enbl_cp_driver (Level 3)

センタープレーンに接続される Sun Enterprise 10000 システム ASIC には、センタープレーンドライバ用に、ループバック制御と、別個に制御される割り込み許可の両方が備わっています。通常、POST は ASIC の設定を通じて、チップがループバックにあるときはドライバが無効になり、ループバックにないときはドライバが有効になるようにします。ボード信号を調べるためには、ループバックのときでも、アドレスバスとデータバスに接続されたセンタープレーンコネクタを駆動することが必要になります。このコマンドを使用することによって、ASIC がボードループバックに設定されている場合でも、ASIC のセンタープレーンドライバはすべて有効になります。pc_loopback_lbus_enbl も参照してください。

gamux_seq_err_enbl (Level 2)

プロダクションマシンには組み込まれない、初期の GAARB リビジョンでは、シーケンスエラーを引き起こす設計上のバグが存在します。1つのドメインで何か別の理由でアービトレーション停止が発生すると、GAMUX ASIC がこのシーケンスエラーを検出します。この種のシーケンスエラーはグローバルエラーとみなされるため、このバグは他のすべてのドメインでアービトレーション停止を引き起こします。

デフォルトでは、rev 1 または rev 2 の GAARB ではこのシーケンスエラーの検出が無効になり、これら以外の GAARB では有効になっています。gamux_seq_err_enbl コマンドはこれを明示的に設定します。GAARB のリビジョン番号に関係なく、0 を指定するとエラー検出は無効になり、1 を指定すると有効になります。

interactive_use_postrc_skips (Level 2)

対話モード (コマンド行オプションの -i) では、.postrc ファイルの skip_phase コマンドや skip_test コマンドは無視され、ユーザーは各フェーズの実行または省略を選択することができます。ただし、場合によってはこうしたデフォルトの設定が不便な場合もあります。何十ものフェーズやテストを対話的に逐一省略するのは面倒です。このコマンドによって、POST はユーザーに問い合わせることなく skip_phase コマンドまたは skip_test コマンドを使用するようになります。

no_skip_phase_covers_npb コマンドも参照してください。no_skip_phase_covers_npb コマンドで記述されている skip は、このコマンドに関しては .postrc ファイルに入っているものとみなされます。

interconnect_MHz *float* (Level 2)

指定したシステムインターコネクト周波数 (MHz) を使用します。SNMP サーバーを通じてこの周波数を測定・報告する制御ボードがないテスト環境では、この値を使用して、システム設定のパラメタ (特にメモリーのタイミング) を計算します。SNMP 周波数の測定値を使用する制御ボードが装着されているシステムでは、この値を指定すると、測定値がこの値の 0.5% の範囲内であることを POST が要求するようになります。この値を指定しない場合は、測定値は SNMP サーバーから読み取られるターゲットのインターコネクト周波数の -3.0 ~ +0.5% の範囲内である必要があります。
`proc_freq_ratio` も参照してください。

iopc_rev_5_ok (Level 2)

リビジョン 5 の PC ASIC には、システムボードの入出力位置、PC 番号 2 で ASIC を使用したときに発生するバグがあります。アプリケーションによっては、ソフトウェア回避策がインストールされているかどうかに関係なく、このバグは問題にはならないものがあります。通常、POST はこの種の ASIC を拒否します。ただし、この種の ASIC の存在がこのコマンドで明示的に指示されている場合は、受け付けます。

jtag_lock_trace (Level 2)

JTAG ボードとリングのロックおよびロック解除に関して、POST は冗長 (*verbose*) デバッグメッセージを出力します。

isolated_sysboard (Level 2)

POST はセンタープレーン ASIC の欠如または損傷を無視します。通常は、こうした欠如や損傷によってプログラムは即座に失敗し、POST はボードをループバックから取り出せなくなります。このコマンドは、センタープレーンが用意されていない開発環境で使用します。

jtag_lock_timeout_secs *value* (Level 2)

SSP コアサービスから JTAG リソースのロックを取得するためのデフォルトタイムアウト値を変更します。

level *level* (Level 1)

コマンド行での同等機能: -l

指定されたレベルによって、実行するテストの程度を特定します。レベルを高くするとテストは詳細になります。有効値は 7 ~ 127 です。hpost -?level を入力すると、簡単な説明が表示されます。

このコマンドはエンドユーザーを制御するのに適しています。ただし、レベルを通常よりも低く設定することは避けてください。レベルを通常よりも低くすると、「未検出のハードウェアの問題をあまり多く抱えていないシステムだけを起動する」という本来の目的にそぐわないこととなります。「起動時間が長くなっても、詳細な診断を実行したい」という場合には、デフォルト (hpost(1M) の -l オプションを参照) よりも高いレベルに設定します。

logfile [*path*] (Level 1)

コマンド行での同等機能: -g

画面のログファイルを作成します。デフォルトのファイル名は postmdd.hmmm.log であり、デフォルトのディレクトリは \$SSPVAR/adm/\$SUNW_HOSTNAME/post です。デフォルトのパスを使用した場合に、\$SSPVAR/adm/\$SUNW_HOSTNAME は存在しているが、デフォルトのディレクトリが存在していない場合は、POST が post のサブディレクトリを作成して、ログを格納します。

path で指定されたファイルを開いて追加書き込みができる場合は、POST はログをそのファイルに格納します。*path* が指定されていない場合で、ディレクトリとして使用されている *path* のデフォルトのファイルを開くことができる場合は、POST はそこにログを格納します。*path* が指定されておらず、しかもデフォルトのファイルを開くことができない場合は、POST はログファイルをカレントディレクトリに (デフォルトの名前を付けて) 作成しようとします。

ログファイルは追加書き込みができるように常に開いており、開閉時にはタイムスタンプが付きます。したがって、呼び出し側のスクリプトやプログラムは、POST の一連の実行によって作成されるログを受け取るときに、単一のファイルを指定することができます。名前の重複による情報の喪失を防ぐことができます。

このコマンドは SSP のディスクスペースを消費する以外には特に害がないため、レベル 1 となっています。ただし、プロダクション環境では通常、POST の出力は (-s を使用した bringup(1M) スクリプトによって) syslog に格納されます。プロダクション環境でこの種の情報を格納するには、syslog のほうが適切です。

```
logger_level{default|emerg|alert|crit|err|warn|
notice|info|debug} (Level 2)
```

POST はライブラリが作成した liblogger メッセージをすべて取得し、POST の表示とログに組み込みます。POST はさらに、デフォルトのロガーレベルを **WARNING** に変更します。このため、POST の表示内容は外部から制御されなくなります。このコマンドを使用すると、POST は環境から引き継いだデフォルトのロガーレベルを変更せず、通常のロガーによる制御を可能にします。または、ロガーレベルを任意の値に指定できるようになります。

使用する .postrc ファイルがローカルディレクトリにない場合は、POST はこのコマンドを無視し、警告メッセージを表示しません。

```
logical_memboard_swap slot1 slot2 (Level 3)
```

名前が付いている 2 つの物理システムボードの論理メモリー番号を交換します。物理メモリーのアドレスは、**8GB × logical board number** (**8GB × 論理ボード番号**) を開始点として POST によって割り当てられます。デフォルトでは、論理メモリーボードから物理システムボードへの割り当ては **1-1** です。時間の経過に従ってスワップが発生するかもしれません。たとえば、動的再構成ボード取り外し操作の結果、あるいは POST によるソート

(**no_memboard_sort** コマンドを参照) の結果としてスワップが発生します。プロダクションシステムの **SSP** では、この物理システムボードと論理メモリーボードの割り当ては **SNMP MIB** に格納され、POST の実行ごとに読み込まれます。サンの技術者はこのコマンドを使用して、**SNMP** が実行されていない試験環境でデフォルト以外の割り当てを試すことができます。

これらのコマンドに数の制限はありません。スワップは要求された順に発生します。2 つの引数には必ず異なるシステムボード番号を指定します。

このコマンドが有効になるのは、カレントディレクトリ内の .postrc ファイルで使用された場合だけです。使用する .postrc ファイルがローカルディレクトリにない場合は、POST はこのコマンドを無視し、警告メッセージを表示します。

このコマンドによって発生したスワップは、**SNMP** から取得したマップ情報によって無効にされ、無視されます。ただし、**no_snmp_memmap** または **no_snmp_access** コマンドによってマップ情報へのアクセスが禁止されている場合には、こうした変更は行われません。**SNMP** から取得したマップを使ってこの種のスワップを実行することはできません。両者は相互に排他的です。

mc_gabfifo_hold_depth 6-bit value (Level 3)

このフィールド (定数レジスタビット [17:12]) の通常の MC ASIC 設定を無効にします。

mc_prescale_disbl (Level 2)

すべてのタイムアウトを $2^{**}10$ システムクロックによってプレスケールさせる通常の MC ASIC 設定を無効にします。このフィールドはタイムアウト選択レジスタビット 23 です。

mc_store_match_to 5-bit value (Level 2)

このフィールド (タイムアウト選択レジスタビット [9:5]) の通常の MC ASIC 設定を無効にします。

mc_timing_reg reg value (Level 3)

指定したメモリータイミング制御レジスタ [3:0] の通常の MC ASIC 設定を無効にします。レジスタ 0 はリフレッシュタイミング、レジスタ 1 と 2 はメモリータイミング 1 と 2、レジスタ 3 は DIMM ワイヤタイミングです。

mc_xmit_req_to 5-bit value (Level 2)

このフィールド (タイムアウト選択レジスタビット [4:0]) の通常の MC ASIC 設定を無効にします。

mc_extra_data_to 5-bit value (Level 2)

このフィールド (タイムアウト選択レジスタビット [14:10]) の通常の MC ASIC 設定を無効にします。

mem_board_interleave_ok (Level 1)

POST がメモリーを設定するときに、同じメモリーサイズの 2 つのボードを、共通のアドレス範囲の中で 256 バイトブロック単位でインタリーブできるようにします。この結果、インタリーブしない場合に比べて、個別のボードの共通のアドレス範囲は 2 倍になります。

インタリーブによって、大きなブロックとなる連続データへのメモリーアクセスが 2 つのメモリーコントローラに分散され、性能が向上します。ただし、2 つのボードをインタリーブすると、動的再構成 (DR) が制限されることがあります。特に、インタリーブされたボードの DR 切り離しが困難になります。

デフォルトでは、POST はボードをインタリーブしません。このコマンドを使用することで、同じメモリーサイズの 2 つのボードをインタリーブすることが可能になります。

```
memboard_sort_order {low_to_high | high_to_low}
(Level 2)
```

このドメイン内のシステムボードにメモリーの物理アドレスを割り当てる順番を指定します。low_to_high を指定すると、小さい番号のシステムボードに低いアドレス、大きい番号のシステムボードに高いアドレスが割り当てられます。high_to_low を指定すると、その逆の順序でシステムボードにアドレスが割り当てられます。

memboard_sort_order コマンドを使用しない場合には、当該ドメインの SSP の SNMP MIB に指定された Solaris オペレーティング環境のバージョンを調べることによってアドレスの順序が決定されます。Solaris 7 よりも前のバージョンの場合には、low_to_high オプションが使用され、Solaris 7 オペレーティング環境で初めて high_to_low が使用されます。

no_snmp_access コマンドにより、SNMP MIB へのアクセスが抑制されている場合は、デフォルト値は low_to_high になります。

この順番付けの目的は、一番小さい番号のボードにカーネルメモリーを配置することにより、オペレーティングシステムのバージョンに関係なく、複雑なシステムを動的再構成 (DR) 用に最適化することです。

no_memboard_sort および no_memboard_fullsort コマンドも参照してください。

```
memboard_swap_threshold_mbytes mbytes (Level 2)
```

このコマンドは、ドメイン内の最小番号の物理ボードに必要な最小メモリー容量を定義します。これは、該当ボードの物理アドレスが、ドメイン内の最大メモリーを持つボードの物理アドレスとスワップされることを避けるためです。ソートの説明およびソートを無効にする方法については、no_memboard_sort コマンドの項を参照してください。

スワップの目的は、動的再構成の「ケージ化されたカーネル」をこの単一ボードに読み込むのに必要なメモリーを、ドメイン内の最初の連続物理アドレス部分に確保することにあります。これによって、任意のボードを DR 切り離しできる可能性が増加します。ただし、最初のボードに必要な最小メモリーが備わっている場合は、より大きなメモリーを持つボードが他にあったとしてもスワップを要求しないようにすることができ、こうした不必要なスワップを避けることができます。

この非スワップしきい値のデフォルト最小値は 512MB です。このコマンドを使用することによって、現場での「ケージ化されたカーネル」の特性の変化に応じて、しきい値を大きくすることができます。最大値は 4096MB(4GB) であり、これは最大容量の DIMM を使用して 1 枚のボードに搭載できるメモリーの総量です。

memchunk_page_truncate #_of_8KB_pages_per_chunk (Level 3)

OBP に渡されるメモリーチャンクのリストで宣言され、設定の最終フェーズでクリアされるメモリーのサイズを制限します。この値より大きい連続したチャンク（通常は 1 つのボードまたはインタリーブされた 2 つのボードの物理メモリー）は、宣言されたサイズにまで切りつめられます。このコマンドはメモリーコントローラの設定には影響せず、OBP に対して宣言されクリアされるメモリーのサイズを決めるだけです。設定済みで初期化されていないメモリーの一部を処理するデバッグ環境では、時間を節約するために、このコマンドを使用します。このコマンドはエミュレーション環境で使用するのに適しています。

memtest_limit [base] limit (Level 2)

POST によってテストされる各 DIMM グループのメモリーのサイズを制限します。引数を 1 つだけ指定した場合は、その引数が制限値 (limit) として使用され、基本値 (base) は 0 となります。どちらの値も (PA 0 でインタリーブなしに設定されたかのように) 各グループ内のバイトオフセットを表します。一般に、「インタリーブなしの設定」という想定は、完全に設定されたマシンの、所定のグループに割り当てられた物理アドレスには対応しておらず、メモリーテストの際に使用される物理アドレスにも対応していません。特にメモリーテストでは、バンクが 4 つともインタリーブされたものと想定されます (4 つのバンクの一部にメモリーが装着されていない場合も同様です)。各プロセッサは、テストしているバイト数の 4 倍のアドレス範囲をテストします。ただし、テストされるのは、256 バイト中の 64 バイト (これらはすべて同じ DIMM グループにあります) だけです。ここで指定する制限値は、テストされる各グループに適用されます。メモリーは 64 バイトのブロックでテストされるため、どちらの引数も 64 の倍数にする必要があります。どちらの引数の場合も、最大値は 0x40000000 (1GB) であり、サポートされている最大の DIMM グループのサイズに相当します。limit は base よりも大きくする必要があります。base が物理グループのサイズよりも大きい場合は、そのグループはテストされません。

new_cid_rev component_id (Level 2)

JTAG の完全性テストの際に、POST はすべての ASIC (および CID を持つその他のチップ) のコンポーネント ID を読み込みます。POST はこれらの ID を有効な CID テーブルと照合して、チップを検査します。通常、CID がテーブルに一致しない場合には、POST は該当チップを拒否します。ただし、不一致がリビジョン番号 (8 桁の CID の最上位 16 進数) だけであり、しかも実際のリビジョン番号の方がテーブル内のリビジョン番号より高い場合、POST は該当チップを受け入れ、*up-level* チップに関する警告メッセージを表示します。

このプロセスによって、POST パッチの配布を実行することなく、ハードウェアのアップグレードに対する対応能力が高まります。また、「POST がすべてのハードウェアのすべての機能と条件を認識していない可能性があるため、ハードウェアの問題が発生した場合には、チップのバージョンとの関連がまず疑われる」という意味のメッセージが表示されます。

up-level メッセージは警告にすぎませんが、ユーザーが不安を感じる可能性があります。new_cid_rev コマンドの目的は、CID テーブルに項目を追加して、*up-level* 警告メッセージを出さないようにすることです。postrc ファイルにはこのコマンドの複数のインスタンスが含まれていることがあります。POST がチェックする値のリストは、*up-level* 警告メッセージが出されるときにのみ作成されます。したがって、new_cid_rev コマンドを使用して新しいベース CID を宣言することはできず、*down-level* CID リビジョンを受け入れ可能であることを宣言することもできません。このコマンドが有効になるためには、追加する CID のベース部分が、いずれかのチップに対するテーブル内の、少なくとも 1 つのベース CID と一致している必要があります。したがって、new_cid_rev コマンドで、テーブルに追加するチップのタイプを指定する必要はありません。

サービスプロバイダがこの機能を利用して *up-level* 警告メッセージを抑止するのは、サンの技術者から「該当 *up-level* チップは POST にコンパイルされている古いリビジョンと実際に同等である」という通知を受け取った場合に限られます。

通常、*component_id* は 8 桁 (32 ビット) の 16 進数で指定します。*component_id* は、有効な CID としての条件 (最下位桁が 1 である) を満たしているかどうかを確認するためにチェックされるだけです。

no_asic_config_check (Level 3)

システム設定の際に JTAG によって書き込まれた ASIC CSR の通常の読み取りと比較を省略します。開発環境でシステムのシミュレーションを行っているときに有用なコマンドです。

no_bbc_error_check (Level 2)

通常、POST は、主要なフェーズを終えるときに、PC 内のブートバスのエラー状態レジスタをチェックし、ブートバスパリティまたはブートバスアクセスのエラーが発生していないかどうかを調べます。この種のエラーが発生すると、PC は失敗します。このコマンドは、このレジスタのチェックを実行しないようにします。

no_bbsram_clear (Level 2)

通常、POST は BBSRAM をすべて書き込み、初期化されていないアドレスにコードがアクセスしたときのパリティエラーを防止します。特別なブートバス SRAM テストファイルとの干渉を避けるために、これはシグニチャーブロックの必要性を宣言されたファイルの最初のコードダウンロードで行われます。ダウンロードの時間を節約するために、POST はコードやデータが読み込まれていない BBSRAM セグメントにだけ、1 回限り書き込みます。このコマンドはこの書き込みを抑止します。したがって、コードが明示的に読み込まれていない BBSRAM 領域は、空白のまま残されます。

no_check_chain_length (Level 2)

JTAG の完全性テストの際に、POST はチップのコンポーネント ID がすべて標準の 32 ビット長であるかどうかをチェックします。この単純かつ有用なテストは、チップの JTAG 走査接続の信頼性を保証します。このコマンドを使用することによって、POST はチェーン長のテストを省略します。JTAG 完全性テストフェーズの検査は引き続き実行されます。

no_dtag_ecc_ce_scrub (Level 2)

Dtag ECC モードに設定されているリビジョン 2 CIC ASIC は、Dtag SRAM アクセス時に検出されたシングルビットエラーを訂正します。通常、訂正されたデータは SRAM にも書き戻されます。「スクラッピング (scrubbing)」と呼ばれるこの機能は、Dtag SRAM の訂正可能なエラー (CE) を解決し、その後続く Dtag SRAM アクセスの間に循環する CE を除去します。この属性を使用すると、POST は、CIC ASIC 設定時にスクラッピング機能を無効にするようになります。

no_dumpfile_on_stop (Level 2)

テストの際にアービトレーション停止またはレコード停止が検出されると、POST は「興味深い」システム状態のダンプファイルを作成します。このコマンドは POST のこの通常の動作を抑止します。

このコマンドを使用しない場合は、POST はシステム状態をダンプし、「興味深い」状態でないボードを省略するように編集してから、ダンプを解析します。「興味深い」状態のボードが見つからない場合は、ダンプファイルを作成せずに、テストを続行します。たとえば、メモリーを原因とするデータ ecc エラーのレコード停止では、ダンプファイルが作成されずに、テストが実行されます。dump_pathname も参照してください。

no_dumpfile_on_stop が .postrc に存在せず、no_stop_analyze が .postrc に存在する場合は、POST はダンプファイルを作成するだけで、テストを続行します。ダンプファイルの編集や解析は行われません。

no_final_flush_reset (Level 2)

通常、POST は PC ASIC 内のレッドリストに入っていないプロセッサのマスターキューをすべてフラッシュしてから、最終設定の最後で各プロセッサをリセットします。この結果、POST のスピン待ちループからのステールフェッチを実行しなくても、OBP コードを BBSRAM に容易にダウンロードできるようになります。no_final_flush_reset を使用すると、プロセッサはスピン待ちを BBSRAM 内の POST の最終設定コードに残します。これは最終設定のプロセッサ状態を事後解析する上で役に立ちます。

no_jtag_locks (Level 3)

通常、Sun Enterprise 10000 の制御ボードの JTAG 通信機能を使用する SSP ソフトウェアは、通信先の JTAG リングをロックして、複数の SSP アプリケーション (POST、obp_helper(1M) など) が相互に干渉することを防止します。このロックは、SSP ライブラリに用意されているセマフォベースのサービスを通じて行われます。

このコマンドを使用した場合は、POST はこうしたロックサービスの使用を省略します。ある種の設計開発環境では便利な機能です。

no_lockfile (Level 3)

Sun Enterprise 10000 の同一ホストドメイン上で複数の POST プロセスを実行できるようにします。通常、POST はロックファイル \$SSPVAR/adm/\$SUNW_HOSTNAME/hpost.lock を作成して、POST の複数のインスタンスを防止します。複数のプロセスを実行すると、相互に干渉して、どちらの POST プロセスも失敗するおそれがあるためです。

通常の SSP 環境では、POST はデーモンモードで実行されます。POST の実行は常に明白であるとは限らず、開発環境では POST プロセスがユーザーが開始した POST プロセスと衝突するといったことがよくあります。

ただし、場合によっては、無効なロックファイルによって POST の実行が拒否されるよりは、2つの POST プロセスを実行した方が安全とみなされることもあります。この場合、ロックファイルの作成に失敗したら、POST はそのロックファイルが他方の POST プロセスによって作成されたものであることを確認しようと試みます。この確認に失敗した場合、POST は下記の説明のとおり、-f モードに入り、残存しているロックファイルを削除します。このプロセスによって、異常終了した POST プロセスが残したロックファイルから自動的に回復することが可能になります。

no_lockfile コマンドはロックファイルの機能 (ファイル作成とファイル削除の両方の機能) を完全に無効にします。

このコマンドはコマンド行オプション -f とは大きく異なります。-f を指定した場合、POST はロックファイル作成を試みるのではなく、作成に失敗してもプロセスを継続するだけです。また、作成に失敗した場合も、終了時にはロックファイルを削除しようとします。-f の目的は、アクティブな POST プロセスに間違って付加された無効なロックファイルから回復することにあります。

no_mc_hardreset (Level 2)

メモリーコントローラ (MC) ASIC にはいくつかの機能が備わっていますが、中でも重要なのはメモリーのリフレッシュです。この機能は、通常のリセットの影響を受けません。これらの機能をクリアするのはハードリセットだけです。ハードリセットは、電源投入時のリセットの際に、JTAG TAP リセットピンによって行われるか、あるいは JTAG 設定レジスタ内の特別なハードリセット制御ビットによって行われます。LBIST が実行されているときには特に、MC をハードリセットを必要とする状態にしておくことで、MC を使用可能にすることができます。

通常 POST は、リセットの初期段階で、すべての MC ASIC に対してハードリセットを実行しますが、このコマンドを使用することによって、ハードリセットが省略されます。これはある種のテスト環境では便利な機能です。POST が特別な再設定モード -z で実行されているときにも、ハードリセットは省略されます。

no_memboard_fullsort (Level 2)

物理システムボードのメモリーの物理アドレスの完全なソートを抑止します。この `no_memboard_fullsort` コマンドが呼び出されると、システムボードへの物理アドレスの割り当てに SSP SNMP ConfMemMap MIB 要素が修正されずに採用されます。ただし、`no_memboard_sort` コマンドの項で説明されているスワップは例外です。

この `no_memboard_fullsort` コマンドが呼び出されない場合は、このドメイン内のアドレスが `memboard_sort_order` コマンドの項で説明した順序で完全にソートされます。次にこのソート順は、`no_memboard_sort` コマンドの項で説明する、生じ得るスワップに従います。

no_memboard_sort (Level 2)

デフォルトでは、複数のボードで実行されている POST は、各ボードのメモリーサイズに基づいてボードを部分的にソートします。このソートの目的は、ドメイン内のカーネルメモリーの物理メモリーアドレスが、最大のメモリーサイズを持つボードに割り当てられるように、ドメインを設定することにあります（最小ボードのメモリーサイズがしきい値を上回る場合はこの限りではありません）。これによって、

memboard_swap_threshold_mbytes コマンドの項で説明したように、動的再構成のいくつかの機能が最適化されます。

このソートを実行するには、ドメイン内でカーネルメモリーを持つ論理ボードとこれより多くのメモリーを持つドメイン内の別のボードの間で、単一の物理・論理マッピングスワップが必要になります。マッピングの詳細は、**logical_memboard_swap** コマンドを参照してください。物理・論理マッピングは SSP の SNMP エージェントによって保持されているため、スワップが必要になった場合には、POST が SNMP への書き込み操作を行い、スワップを知らせます。

no_memboard_sort コマンドは、ソートと SNMP への書き込みアクセスの両方を抑止します。**no_snmp_memmap** または **no_snmp_access** が起動されているときは、暗黙的に抑止されます。**no_memboard_fullsort** も参照してください。

no_memory_ok (Level 2)

オペレーティングシステムはメモリーがないと起動できないため、メモリーのサイズは、設定の評価に使用する性能指数 (FOM) の一要素となっています。メモリーがないときは 0 FOM になります。ただし、ボードの立ち上げの際には、メモリーが装着されていない場合でも POST によるテストを続行する場合があります。このコマンドは、POST に対して、実際にはメモリーが存在しない場合でも、1 単位 (現時点では 64MB) のメモリーが存在するものとして FOM を計算するように指示します。

プロセッサモジュールと入出力機構の他の機能については、実際のメモリーのテストが必要です。メモリーなしでは他の要素がテストされないため、メモリーなしでボードテストを実行することを標準としないでください。ただし、状況によってはこのコマンドが役に立つこともあります。

no_non_proc_boards (Level 1)

プロセッサなしのシステムボードを不可にします。デフォルトでは、プロセッサの障害によって、オンボードのメモリーや入出力機構が使用不可になることを防ぐために、プロセッサなしのボード (NPB: Non-Processor Board) は許容されています。通常は、こうした設定で特に問題はありません。

しかし、場合によっては、NPB が望ましくないことがあります。このコマンドを使用することによって、プロセッサなしのボードは、入出力機構やメモリーを装備していても、設定から削除されることとなります。

no_obp_handoff (Level 2)

最終設定のフェーズで、ブートプロセッサのブートバス SRAM に post2obp 構造を作成しません。このコマンドを使用することによって、POST は、ブートプロセッサ番号 (0 ~ 63) の代わりに POST_EXIT_NOCONFIG で終了します。

no_poll_board_arbstop (Level 3)

アービトレーション停止とレコード停止に対する POST の通常のポーリングを抑止します。POST は JTAG を介してローカルアドレスアービター (LAARB: Local Address Arbiter) を読み取り、ポーリングを実行します。このコマンドを使用することによって、別の方法で (ほとんどの場合、タイムアウトによって) ストップが検出されます。エラーの実際の原因の代わりにプロセッサが失敗し、回復の機能は限定されます。

no_poll_timeouts (Level 3)

ホストプロセッサの RPC サービスのタイムアウトをすべて無効にします。サンの技術者による POST のデバッグで、ホストプロセッサの常駐コードにブレークポイントを設定して実行しているときには、このコマンドが役に立ちます。poll_timeout_mult と比較してください。

no_procs_ok (Level 2)

オペレーティングシステムはプロセッサがないと起動できないため、プロセッサの数は、設定の評価に使用する性能指数 (FOM) の一要素となっています。プロセッサなしのときは 0 FOM になります。ただし、ボードの立ち上げの際には、プロセッサが装着されていない場合でも、POST によるテストを続行する場合があります。このコマンドは、POST に対して、実際にはプロセッサが存在しない場合でも、1 個のプロセッサが存在するものとして FOM を計算するように指示します。プロセッサがない場合、POST が実行する機能はごく限られます。しかし、JTAG ASIC 設定の実行は可能なため、特定の立ち上げ環境においては役立ちます。

no_scard_probe (Level 2)

欠落している *scard* の検査をしません。一般に **SBus** または **PCI** バスいずれかの入出力アダプタカードを *scard* と呼びます。このコマンドが呼び出されると、特定の入出力モジュールに設計上定義されたすべての入出力カードスロットにカードが装着されたとみなされます。この情報は、**POST** が実行する性能指数 (**FOM**) 計算に影響を与え、**OBP** に報告されます。これにより **POST** によりカードの装着が宣言されたすべてのスロットの検査が可能になります。

SBus カードの場合にも、適切な検査で *skip_test* コマンドを使用することにより、検査を省略することが可能です。ただし、**PCI** 入出力モジュールの場合には検査が行われません。**PCI** 入出力モジュールの検査は **SSP** から **JTAG** によって実行されます。

no_scards_ok (Level 2)

オペレーティングシステムは入出力機構がないと起動できないため、**SBus** カードの数 (より一般的には、カードを装着した入出力スロットの数) は、設定の評価に使用する性能指数 (**FOM**) の一要素となっています。**SBus** カードなしのときは **0 FOM** になります。ただし、ボードの立ち上げの際には、**SBus** カードが装着されていない場合でも、**POST** によるテストを続行する場合があります。このコマンドは、**POST** に対して、実際には **SBus** カードが存在しない場合でも、1 個の **SBus** カードが存在するものとして **FOM** を計算するように指示します。

no_skip_phase_covers_npb (Level 2)

可能な場合には、POST は他のオンボードリソースを使用して、ボードのループバック内のシステムリソースをテストします。これによって、複数のテストを別々のシステムボードで同時に実行できるため、失敗の原因を特定する POST の能力が向上し、テストの実行が高速になります。

システムボードに空白がある場合、あるいはオンボード障害が発生した場合には、POST はオフボードリソースを使用する必要があります。たとえば、良好なプロセッサを装備していないボード (**Non-Processor Board: NPB**) はオフボードのプロセッサを使用してメモリーと入出力機構をテストします。良好なメモリーを装備していないボード (**Non-Memory Board: NMB**) はオフボードのメモリーを使用して、プロセッサと入出力機構をメモリーに対してテストします。単一のプロセッサを備えたボードは、他のボードのプロセッサに対してクロスコール割り込みテストを実行します。skip_phase コマンドを使用し、オンボードメモリーテストなどの基本テストのフェーズを省略した場合は、技術者のほとんどはメモリーテストがすべて省略されたものとみなします。しかし、続いて NPB メモリーのテストが行われ、POST の後の段階でオンボードメモリーのテストが行われると、こうした想定は混乱の原因になります。したがって、POST のデフォルトの動作では、基本フェーズが省略されたときには、同じリソースをオフボードでテストするすべてのフェーズが自動的に省略されることとなります。

これは一般的には望ましい動作ですが、困る場合もあります。センタープレーンのインターコネクトのストレステストなどの場合には、低速ではあっても、すべてのテストをループバックから実行する方法が適しています。あるいは、基本テストを省略しても、特定の問題の障害追跡のために NPB テストを実行するという場合もあります。これを可能にするために、

no_skip_phase_covers_npb コマンドは「基本フェーズを省略したときにはオフボードフェーズをすべて省略する」というデフォルトの動作を無効にします。

このコマンドは NPB という省略形を使用していますが、実際には同様のすべてのオフボードテスト (たとえば、単一プロセッサのボード用の NMB テストや xcall テスト) に影響します。

また、省略はフェーズレベルでのみ適用されることにも注意してください。skip_test コマンドによって基本フェーズのテストがすべて省略されている場合は、そのフェーズが省略されたものとはみなされません。

no_snmp_access (Level 2)

通常、POST は SSP の SNMP エージェントにアクセスして、実行に必要な各種の情報を取得します。このコマンドはこのアクセスを抑制します。このコマンドは SNMP エージェントが実行されていないサンの試験環境をサポートすることを目的とし、カレントディレクトリの .postrc ファイルで使用されたときにのみ有効です。 .postrc の検索パスの後続の項目は *platform_name* を必要とし、 *platform_name* は SNMP エージェントから取得する必要があります。使用する .postrc ファイルがローカルディレクトリにない場合、POST はこのコマンドを無視し、警告メッセージを表示します。

SNMP から取得する情報の多くは、.postrc ファイルまたはプログラムのデフォルト値によっても提供されます。

interconnect_MHz、proc_freq_ratio、domain、
logical_memboard_swap、platform_name、
no_snmp_freq_read の各コマンドを参照してください。

no_snmp_invalidate_bootproc (Level 2)

デフォルトでは、POST の起動に成功すると、範囲 [0,63] のブートプロセッサ番号が返され、POST はホストとの対話を開始する前に実行していたドメインの SNMP のブートプロセッサ値を無効にします (例外: -H モードの動的再構成では POST は SNMP 値を無効にしません)。SNMP 値を無効にすることによって、このドメイン内で他の SSP 監視プログラム (特にイベント検出デーモン (EDD)) がアービトレーション停止などのイベントをチェックすることがなくなります。他のプログラムによるチェックは hpost(1M) の実行に干渉するおそれがあります。

no_snmp_invalidate_bootproc コマンドはこの無効化を抑制します。no_snmp_access を使用することによって、no_snmp_invalidate_bootproc も使用されたことになりません。

POST は SNMP エlement を有効な値に設定するわけではありません。POST が行うのは、SNMP 値を無効にすることだけです。通常、有効なブートプロセッサ値の設定は bringup(1M) によって行われます。したがって、POST が bringup(1M) 以外によって実行されている場合、SNMP bootproc MIB は無効化されたままになります。

no_snmp_memmap (Level 2)

通常、POST は SSP の SNMP エージェントにアクセスして、物理メモリーボードと論理メモリーボードのマッピングを取得または変更します。このコマンドはこのアクセスを抑止します。

no_snmp_access が起動されている場合は、このコマンドも使用されたこととなります。このマッピングの詳細は、logical_memboard_swap コマンドを参照してください。このコマンドが有効になるのは、カレントディレクトリの .postrc ファイルで使用された場合だけです。使用する .postrc ファイルがローカルディレクトリにない場合は、POST はこのコマンドを無視し、警告メッセージを表示します。

no_snmp_freq_read (Level 2)

通常、POST は SSP の SNMP エージェントにアクセスして、プラットフォームインターコネクトとプロセッサ周波数の値を取得します。このコマンドはこのアクセスを抑止します。

no_snmp_access が起動されている場合は、このコマンドも使用されたこととなります。no_snmp_freq_read または no_snmp_access が使用された場合は、POST はプラットフォームインターコネクトとプロセッサ周波数の値を .postrc コマンドの interconnect_MHz と proc_freq_ratio または組み込まれているデフォルト値から取得します。

no_stop_analyze (Level 2)

通常、テスト中にアービトレーション停止またはレコード停止を検出した場合は、POST は各種の ASIC の状態を読み取り解析し、障害と判別したコンポーネントを拒否します。このコマンドはこうした解析を抑止します。

pc_color_freq_code (Level 3)

PC ASIC は 2 ビットの設定パラメタであり、アービトレーションカラー値 (アービトレーション公平アルゴリズムの属性) の変更を使用する周波数を制御します。通常はデフォルト値が使用されます。サンの技術者はこのコマンドを使用して、このパラメタの値を試します。このフィールドは設定レジスタのビット [19:18] です。

pc_errmask {0|1 | io0|io1} bit_number (Level 3)

設定の際に、PC ASIC のエラーマスクレジスタの個々のビットを強制的に 0 にし、指定したエラーがアービトレーション停止を引き起こさないようにします。このコマンドには複数のインスタンスとバージョンがあります。マスクされるビットは累積されます。このコマンドに数値 (0 または 1) を付けると、プロセッサ PC だけに適用されます。io (io0 または io1) を付けると、入出力 PC だけに適用されます。0 または 1 はエラーマスクレジスタを指定します。bit_number は 0 ~ 31 の整数です。

pc_grant_to 6-bit pse/value (Level 2)

このフィールド (タイムアウト・保持設定レジスタビット [10:6]) の通常の PC ASIC 設定を変更します。

pc_grant_to_io 6-bit pse/value (Level 2)

このフィールド (入出力 PC 用のタイムアウト・保持設定レジスタビット [10:6]) の通常の PC ASIC 設定を変更します。このコマンドは pc_grant_to コマンドを上書きします。

pc_loopback_always (Level 3)

PC 設定のたびに PC ループバックモードを強制的に実行します。

pc_loopback_lbus_enbl (Level 3)

プロセッサモジュールの初期テストの際に、POST は「PC ループバックモード」と呼ばれる PC ASIC の機能を使用します。このモードでは、PC はトランザクションを CIC を介して送信せずに、内部的にループします。こうすることによって、障害の原因が CIC または PC-CIC 信号にないことが明確になるため、原因の判別が容易になります。

ボードのループバックの場合と同様、PC には PC ループバック用の別個の制御と、CIC へのローカルバス用のドライバインペブルが備わっています。通常、POST はこの設定を通じて、PC ループバックモードでドライバを使用不可にします。このコマンドは、これらのドライバを PC ループバックモードで有効にし、デバッグのときに検査できるようにします。force_enbl_cp_driver コマンドと比較してください。

pc_master_pio_req_limit 2-bit_code (Level 3)

PC 用のこのフィールド (タイムアウト・保持設定レジスタビット [24:23]) の通常の PC ASIC 設定を変更します。入出力 PC では常に 0 です。

pc_master_read_to 6-bit_pse/value (Level 2)

このフィールド (タイムアウト・保持設定レジスタビット [5:0]) の通常の PC ASIC 設定を変更します。ビット [5] (このフィールドの最上位ビット) は、PC 内のマスターの読み取りと割り込み、およびスレーブの応答のすべてのタイムアウトに対する、プレスケールの許可です。

pc_master_read_to_io 6-bit_pse/value (Level 2)

入出力 PC 用のこのフィールド (タイムアウト・保持設定レジスタビット [5:0]) の通常の PC ASIC 設定を変更します。

pc_master_read_to_io コマンドは pc_master_read_to コマンドを上書きします。

pc_pc_hold 5-bit_value (Level 3)

このフィールド (タイムアウト・保持設定レジスタビット [22:18]) の通常の PC ASIC 設定を変更します。

pc_slave_response_to 5-bit_value (Level 2)

指定した 25 ビット値を使用して、このフィールド (タイムアウト・保持設定レジスタビット [31:27]) の通常の PC ASIC 設定を変更します。このタイマーは、マスターの読み取りタイムアウトに対してプレスケールの許可を使用します。pc_master_read_to を参照してください。

pc_slave_response_to_io 5-bit_value (Level 2)

指定した 5 ビット値を使用して、このフィールド (入出力 PC 用のタイムアウト・保持設定レジスタビット [31:27]) の通常の PC ASIC 設定を変更します。このコマンドは pc_slave_response_to コマンドを上書きします。

pc_slave_wr_to 6-bit_pse/value (Level 2)

このフィールド (タイムアウト・保持設定レジスタビット [16:12]) の通常の PC ASIC 設定を変更します。

pc_slave_wr_to_io 6-bit_pse/value (Level 2)

このフィールド (入出力 PC 用のタイムアウト・保持設定レジスタビット [16:12]) の通常の PC ASIC 設定を変更します。

pc_slave_wr_to_io コマンドは pc_slave_wr_to コマンドを上書きします。

pc2_iopc_cic_perr_disable (Level 2)

入出力 PC ASIC による CIC 制御バス上のパリティエラーの検出を無効にします。システム内のいずれかのボードにマジックワイヤー ECO が備わっている場合、こうした指定が必要になります。入出力位置 (ボード上の PC 2) に Rev 4 以前の PC が備わっているボードについては、この条件が該当します。この種の PC は ECO なしでは信頼性を実現できません。

phase_time_report (Level 2)

POST の各フェーズの経過時間を報告します。

platform_name string (Level 2)

`no_snmp_access` コマンドの項を参照してください。
`no_snmp_access` コマンドが起動している場合、`platform_name` コマンドがプラットフォーム名を提供します。通常、プラットフォーム名は SNMP エージェントから取得します。`no_snmp_access` コマンドが起動していない場合、POST は `platform_name` コマンドを無視します。`no_snmp_access` コマンドが起動していて、`platform_name` コマンドが起動していない場合は、POST は `$SUNW_HOSTNAME` をプラットフォーム名として使用します。
このコマンドが有効になるのは、カレントディレクトリの `.postrc` ファイルで使用された場合だけです。使用する `.postrc` ファイルがローカルディレクトリにない場合、POST はこのコマンドを無視し、警告メッセージを表示します。

poll_timeout_mult mult_factor (Level 2)

ホストプロセッサにタスクを割り当てたときに使用するタイムアウト値に、ここで指定した整数値 (1 ~ 100) を乗算します。多くのプロセッサを装備したシステムで非常に冗長なメッセージの出力が有効になっているような場合は、良好なプロセッサがポーリングによって遅延し、「通常の許容時間を超過した」というオーバーヘッドを表示して、失敗してしまふことがあります。このコマンドはこうしたタイムアウトをすべて延長します。現場でのタイムアウト値が小さ過ぎる場合にも、このコマンドを使用することができます。`servmgr_time_report` コマンドと `no_poll_timeouts` コマンドの項も参照してください。

port_idle_use_pause (Level 3)

PORT_IDLE 機能の代わりに、あまり柔軟でない PORT_PAUSE PC ブートバス機能を使用します。

print_all_errors (Level 2)

通常 POST は、特定のテストの最初の 1 つか 2 つのメッセージだけを出し、そのテストを終了します。このコマンドを使用することによって、エラーメッセージをすべて出力して、テストを継続します。

この機能を使用すると、多くのエラーメッセージが出力され、テストのタイムアウトを超過することがあります。この結果、良好なプロセッサまでが失敗したとみなされます。こうした事態を防ぐには、`poll_timeout_mult` コマンドと一緒に使用してください。

proc_freq_check_percent {float_% | off} (Level 2)

JTAG 完全性テストのフェーズで、POST は各プロセッサの最高定格速度を読み取り、制御ボードから供給されるシステム全体のプロセッサクロック速度と比較します。クロック速度と定格速度の差が指定した割合 (`float_%`) を超えると、POST は該当するプロセッサを拒否します。デフォルト値は 0.5% です。引数 `off` を指定すると、POST はこのチェックを行いません。

proc_freq_ratio {2 | 2:1 | 3 | 3:1 | 3:2} (Level 2)

システムインターコネクタ周波数に対するプロセッサ周波数の比率を指定します。プロセッサ周波数を測定し報告する制御ボードが欠如しているテスト環境では、この値を使用して、`post2obp` 構造体にプロセッサ周波数を報告し、プロセッサの最高定格速度をチェックします。

制御ボードが装着されているシステムでは、この値を指定すると、測定された比率がこの値の 0.5% 以内であることが必要になります。この値を指定しない場合は、測定された比率は SNMP サーバーから読み取ったターゲット比率の 0.5% 以内である必要があります。`interconnect_MHz` コマンドの項も参照してください。

proc_runslow_warn_percent[*float_%* | *off*] (Level 2)

通常 POST は、制御ボードの周波数に対するプロセッサモジュールの定格速度を検査し、プロセッサの速度が定格速度以上になっていないことを確認します。`proc_freq_check_percent` を参照してください。またプロセッサの速度が定格速度よりも著しく遅いかどうかも検査します。遅い場合には、制御ボードの設定が間違っていることが考えられます。速度が定格速度よりかなり遅い場合には、考えられる設定ミスについて知らせる NOTE が POST によって表示されます。

NOTE は、制御ボードの周波数がモジュールの定格周波数より `proc_runslow_warn_percent %` 下回る場合に出力されます。デフォルト値は、9% です。このコマンドを使用すると、別の値の設定、検査、および NOTE の表示を完全に無効にすることができます。

proc_timestamp interval_secs (Level 2)

全ホストプロセッサのダウンロード済みプロセッサコードによって作成されるメッセージを出力するときに（これらのメッセージの先頭には {board.module} が表示されます）、メッセージが作成された時刻を示すタイムスタンプを出力します。ただし、特定のプロセッサのタイムスタンプが出力されるのは、そのプロセッサがメッセージを出力し、そのプロセッサに最後のタイムスタンプが押されてから *interval_secs* 以上の時間が経過しているときに限られます。タイムスタンプの形式は {board.module} {day/date/time} です。

interval_secs は秒数を示す整数であり、有効値は 0 ~ 86400 (= 60 × 60 × 24 秒 = 24 時間) です。0 を指定すると、すべてのメッセージにタイムスタンプが付きます。

qt_missrefresh_err_enbl (Level 3)

quickturn コマンドの項を参照してください。quickturn を指定した場合の POST のデフォルトの処理は、MC ASIC 内で捕捉できなかったリフレッシュエラーをマスクすることです。

qt_missrefresh_err_enbl を使用すると、こうしたエラーがアービトレーション停止を引き起こすこととなります。

quickturn コマンドが使用されていない場合は、このコマンドは無視されます。

quickturn KHz Mem_refresh_interval (Level 3)

Quickturn エミュレーション環境でシステムを実行していることを宣言します。KHz はシステムクロックの速度を示し、有効値は 100 ~ 2000 (2 MHz) です。Mem_refresh_interval はメモリーコントローラ設定用のシステムクロックサイクルでのリフレッシュ間隔を示し、有効値は 8 ~ 0xFFFF (4095) です。どちらの引数も必須です。

ポーリングのタイムアウトは、宣言されたシステム周波数に基づいて、通常値の倍数に設定されます。メモリーコントローラは、指定したリフレッシュ間隔を使用するように設定されます。このコマンドを使用すると、他の設定属性も必要に応じて変更されます。

recordstop_poll (Level 2)

アービトレーション停止のポーリングの際に、レコード停止のポーリングも実行します。

redlist_file {*filepath* | none} (Level 2)

コマンド行での同等機能: -R

デフォルト (\$SSPVAR/etc/*platform_name*/redlist ファイル) の代わりに、指定した **redlist(4)** ファイルを使用します。none を指定すると、POST は redlist ファイルを読み込みません。

blacklist(4) と **redlist(4)** を参照してください。

プロダクションシステムで **redlist_file** を使用するときには注意が必要です。POST が標準外の **redlist(4)** ファイルを使用していることは、他の SSP ソフトウェアにはわかりません。

scantool_simulate (Level 3)

JTAG ライブラリ関数の通常の呼び出しを途中でさえぎり、POST 内部で (単純に) シミュレートします。このコマンドは、本物のシステムやハードウェアによってシミュレートされたシステムがない開発環境で、サンの技術者がコードをデバッグする際に有用です。

servmgr_time_report (Level 2)

ホストプロセッサにタスク (テストの実行など) を割り当てると、POST のサーバーマネージャーはタイムアウトを設定します。タイムアウトが経過するまで完了されないタスクは失敗したものとみなされます。**servmgr_time_report** コマンドを使用することによって、サーバーマネージャーは、プロセッサからの報告があった時点で、タイムアウト到達までにどのくらいの時間が残っているかを報告します。適切なタイムアウトを設定する上でこの情報は役に立ちます。

shared_mem brd boardmask (Level 3)

指定したシステムボードの設定において、**boardmask** 内のシステムボードに対して共有メモリーを提供します。**boardmask** の中には、指定したボードのドメインに属するボードを含めることはできません。詳細は、**domain** コマンドを参照してください。CIC 内の共有メモリー BAR/LAR は **shared_mem** コマンドによっては設定されません。通常この設定は、ホストソフトウェアによって行われます。ただし、**shmem_barlar** を参照してください。

shmem_barlar *bar lar* (Level 3)

指定した *bar* の値と *lar* の値を使用して、JTAG CIC 共有メモリの基本アドレスと限界アドレスのレジスタを設定します。このコマンドは、`shared_mem` コマンドによってメモリーを共有すると宣言されたボードに対してのみ適用されます。また、これらすべてのボードに対して使用される *barllar* の値のセットは 1 つだけです。デフォルトでは (およびメモリーを共有しないボードについては)、*bar* は 1 で、*lar* は 0 です (有効範囲外)。メモリーを共有するボードの場合、これら 2 つの値にボードの基本物理アドレス ($\text{mod } 2^{*41}$) を加算した値が、そのボードの *barllar* 設定となります。これら 2 つのレジスタカウントは 64KB 単位で増加し、 $[15:0] = 0$ となります。*bar* および *lar* はどちらも任意の 25 ビット値にすることができ、*bar* が *lar* より大きくなることもあります。

skip_phase *phase_name* (Level 2)

指定した POST フェーズを省略します。1 つのコマンドで複数のフェーズの省略を指定することはできませんが、`.postrc` の中で使用できる `skip_phase` コマンドの数に制限はありません。用意されているフェーズ名を表示するには、`hpost -?postrc` を実行してください。メッセージの冗長 (`verbose`) レベルを高く設定しておくことによって、フェーズ名は進捗メッセージとしても表示されます。`interactive_use_postrc_skips` コマンドと `no_skip_phase_covers_npb` コマンドも参照してください。

skip_test *test_id* (Level 2)

指定したテストを省略します。1 つのコマンドで複数のテストの省略を指定することはできませんが、`.postrc` の中で使用できる `skip_test` コマンドの数に制限はありません。*test_id* は 0 ~ 0x1FF の数値で、POST テストのインデックスおよび識別子として使用されます。冗長 (`verbose`) レベルを高く設定しておけば、テストを実行する前に、テスト名とともにテスト ID も表示されます。このコマンドで使用すべき ID を判別するには、この方法で表示させるのが最も適切です。テストが失敗したときにもテスト ID が表示されます。

このコマンドは `skip_phase` よりも高度な制御を実現します。現場でのテストが失敗した場合に、バグやハードウェア変更を回避する手段としてもこのコマンドは有用です。

ある種の POST テストは内部的に「省略不可」と宣言されています。こうした宣言のチェックは、POST が `.postrc` コマンドを読み込んだときではなく、テスト実行時に行われます。該当するテストに `skip_test` コマンドが適用されている場合は、POST は「このコマンドを無視する」という内容のメッセージを表示します。`interactive_use_postrc_skips` コマンドも参照してください。

sysreset_ignore_redlist (Level 3)

init_reset フェーズでのシステムリセットを省略せずに、物理的には存在しないシステムリソースをレッドリストに入れることを有効にします。このコマンドはある種の開発環境でのみ使用されず、**redlist(4)** を参照してください。

system_red_any_fail (Level 2)

POST が失敗した場合、ホストシステム全体を内部的に失敗させ、レッドリストに入れます。デバッグの際、障害の事後分析にこのコマンドが役に立つことがあります。

このコマンドを使用すると、障害が発生したコンポーネントを **POST** によって管理し設定する可能性が完全に失われます。したがって、通常は、このコマンドの使用はお勧めできません。このコマンドは、**board_red_any_fail** コマンドを極端にしたものであると言えます。

trace_print board proc_module (Level 2)

指定したホストプロセッサからの非常に冗長な (*verbose*)

tprintf トレース出力を有効にします。デフォルトでは、このコマンドはオフになっています。複数のプロセッサのトレース出力を制御するには、このコマンドの複数のインスタンスを使用してください。

verbose level (Level 1)

コマンド行での同等機能 : *-v*

POST が出力する進捗情報の量を制御します。冗長 (*verbose*) レベルを高くすると、より詳細な情報が出力されます。*level* は 0 ~ 225 の整数値です。レベルの詳細は、**hpost -?verbose** を実行してください。

このコマンドはシステム管理者が使用することができます。ただし、冗長レベルを高く設定すると、**POST** の実行が遅くなることに注意してください。極端な場合にはタイムアウトが発生することもあります。

verbose_print board proc_module (Level 2)

個々のホストプロセッサからの冗長な (*verbose*) *vprintf* 出力を制御します。このコマンドは指定した特定のプロセッサに適用されます。複数のプロセッサに適用するには、このコマンドの複数のインスタンスを使用してください。冗長 (*verbose*) レベルが高く設定されている場合は、このコマンドはすべてのプロセッサに対して有効です。

関連項目

bringup(1M), **hpost(1M)**, **blacklist(4)**, **redlist(4)**

名前	redlist - 操作されないシステムリソースのリスト
注意	このファイルの使用については、購入先の技術者と相談してください。このファイルを変更すると、システムを使用できなくなることがあります。
機能説明	<p>\$SSPVAR/etc/platform_name/redlist は ASCII ファイルであり、システム管理者 (またはスーパーユーザー) が SSP からホストシステムの設定を制限するときに使用します。このファイルには、POST が操作してはならないコンポーネントがリストされています。POST はこれらのコンポーネントの状態を変更することはできません。POST は redlist ファイルを読み取ってから、システムを起動するための準備を行い、テストに成功したコンポーネントだけを OBP に渡します。redlist にリストされたコンポーネントは OBP には渡されません。</p> <p>redlist ファイルは非常に強い制限を適用するファイルであり、ごくまれな場合に使用されます (試験環境での特別なデバッグのために使用するのが大半です)。たとえば、Sun Microsystems の技術者は特定のコンポーネントが欠如したハードウェア (フリーズするだけであっても、POST はこのハードウェアとのやり取りを予期しています) を試験するためにこのファイルを使用します。</p> <p>レッドリストに入れられたコンポーネントは、実際にはブラックリストにも入ります。</p> <p>コンポーネントをレッドリストに入れると、機能とパフォーマンスが制限されます。ボード上の何らかのコンポーネントがレッドリストに入っている場合、POST はそのボードをリセットできなくなります。この場合、ある種の障害をクリアするにはボードのリセットが必要であるため、ボード全体が使用不可能になり、システム全体が使用不可能になることさえあります。</p> <p>注意 - ブラックリストで目的を達成できる場合には、レッドリストには入れないでください。</p> <p>redlist ファイルの特性は次のとおりです。</p> <ul style="list-style-type: none"> ■ キーワードには大文字と小文字の区別がありません。 ■ # で始まる行部分はコメントです。 ■ 0x で始まる場合を除き、数字はすべて 10 進数で表記されています。0x で始まる数字は 16 進数です。ただし、[a - f] または [A - F] のいずれかで指定されたボード番号は 16 進数とみなされます。 ■ 各行にはキーワードが 1 つだけ存在します。 ■ 同じキーワードを複数の行で使用することができます。 ■ キーワードには、それぞれ 1 つまたは複数の引数を使用することができます。引数は整数値で表され、複数の整数値はピリオドで区切られます。

キーワード

下記のすべての値の範囲は境界値を含みます。

<code>sysbd board</code>	指定したシステムボードをテストまたは設定しません。 <code>board</code> は 0 ~ 15 の整数です。
<code>proc board.pmod</code>	指定したシステムボード内の指定したプロセッサをテストまたは設定しません。 <code>board</code> は 0 ~ 15 の整数、 <code>pmod</code> は 0 ~ 3 の整数です。
<code>abus abus</code>	指定したアドレスバスをテストまたは設定しません。 <code>abus</code> は 0 ~ 3 の整数です。すべてのシステムボードについて、対応する CIC ASIC がレッドリストに入れられます。下記のキーワード <code>cic</code> を参照してください。
<code>dbus dbus</code>	144 ビットのデータルータのうち、指定した 72 ビットハーフをテストまたは設定しません。 <code>dbus</code> は 0 または 1 の整数です。すべてのシステムボードについて、ローカルデータルータの指定したハーフ部分がレッドリストに入れられます。下記のキーワード <code>ldpath</code> を参照してください。
<code>ioc board.ioctl</code>	指定したシステムボード上の指定した入出力コントローラをテストまたは設定しません。 <code>board</code> は 0 ~ 15 の整数、 <code>ioctl</code> は 0 または 1 です。
<code>scard board.ioctl.slot</code>	指定したシステムボード上の指定した入出力コントローラ内にある、指定した入出力アダプタカードをテストまたは設定しません。 <code>board</code> は 0 ~ 15 の整数、 <code>ioctl</code> は 0 または 1、 <code>slot</code> は 0 ~ 3 の整数です。
<code>mem board</code>	指定したシステムボード上のメモリーをテストまたは設定しません。 <code>board</code> は 0 ~ 15 の整数です。
<code>mggroup board.group</code>	指定したシステムボード上の指定した DIMM メモリーのグループをテストまたは設定しません。 <code>board</code> は 0 ~ 15 の整数、 <code>group</code> は 0 ~ 3 の整数です。

<code>cplane half_centerplane</code>	センタープレーンのうちの指定したハーフ部分をテストまたは設定しません。センタープレーンハーフには、2つのアドレスバスと72ビットのグローバルデータルータが備わっています。 <code>half_centerplane</code> は0または1です。このコマンドの機能は、該当センタープレーンハーフのバスに対して <code>abus</code> と <code>dbus</code> の両方のコマンドを指定した場合と等価です。
<code>pc board.pc</code>	指定したシステムボード上の指定したポートコントローラ ASIC をテストまたは設定しません。 <code>board</code> は0～15の整数、 <code>pc</code> は0～2の整数です。
<code>xdb board.xdb</code>	指定したシステムボード上の指定したデータバッファ ASIC をテストまたは設定しません。 <code>board</code> は0～15の整数、 <code>xdb</code> は0～3の整数です。
<code>cic board.cic</code>	指定したシステムボード上の指定したコヒーレントインタフェースコントローラ ASIC をテストまたは設定しません。 <code>board</code> は0～15の整数、 <code>cic</code> は0～3の整数です (<code>cic</code> は該当ボードのアドレスバスに対応します)。
<code>ldpath board.dbus</code>	指定したシステムボード上の144ビットデータルータのうち、指定した72ビットハーフをテストまたは設定しません。 <code>board</code> は0～15の整数、 <code>dbus</code> は0または1です。

使用例

例1 redlist(4) ファイルの例

```
# Sun Microsystems, Inc.
sysbd 3 5 0xA # ボード 3、5、10 を使用不可にする
sysbd 3 5 A # ボード 3、5、10 を使用不可にする
PROC 4.0 6.2 # システムボード 4 のプロセッサ 0 および
# システムボード 6 のプロセッサ 2 を使用不可にする
ScarD 3.0.1 # システムボード 3 の入出力コントローラ 0 に
# 装着されている入出力アダプタを使用不可にする
mem 2 # システムボード 2 のメモリーをすべて使用不可にする
mlimit 0xE 2.64 # システムボード 14 の DIMM グループ 2 の
# 使用を 64MB に制限する
cIc 1.2 # システムボード 1 の CIC ASIC 2 を使用不可にする
```

関連項目

`hpost(1M)`, `blacklist(4)`

名前	ssp_resource - SSP プロセスのリソースファイル
注意	このファイルは手動では編集しないでください。手動で編集すると、SSP プロセスによるこのファイルの使用と衝突するおそれがあります。
機能説明	<p>ssp_resource は SSP が使用する汎用のリソースファイルであり、下記の情報が収められています。各フィールドはピリオドで区切られます。</p> <p><i>platform_type.resource_name:resource_value</i> <i>platform_type.id_name.resource_name:resource_value</i></p> <p><i>platform_type</i> SSP のプラットフォームタイプです。現在、<i>platform_type</i> は Ultra-Enterprise-10000 だけです。</p> <p><i>resource_name</i> リソースの名前です。このファイル内で他と重複しない名前を指定します。</p> <p><i>resource_value</i> リソースの値です。</p> <p><i>id_name</i> 該当リソースを現在所有している SSP プロセスまたはドメインの名前です。</p>
使用例	<p>例 1 ssp_resource(4) の例</p> <pre style="border: 1px solid black; padding: 5px;">Ultra-Enterprise-10000.confMemAddrMap:000102030405 ... OEOF Ultra-Enterprise-10000.edd.scripts:sysbrdtemp.tcl,centerplanetemp.tcl Ultra-Enterprise-10000.HIGHasic:60000</pre>
ファイル	<p>以下のファイルがサポートされます。</p> <p>\$SSPVAR/.ssp_private/ssp_resource</p>
関連項目	cbs(1M), edd(1M), snmpd(1M)

名前	ssp_to_domain_hosts - ホスト名とドメイン名のファイル
注意	このファイルは変更しないでください。このファイルを手動で変更すると、SSP の動作に影響します。
機能説明	\$SSPVAR/.ssp_private/ssp_to_domain_hosts は、ドメイン名のリストを保守します。下記の「使用例」に示すように、各ドメイン名の右側には、それぞれに割り当てられた SSP ホスト名が表示されます。
使用例	例 1 ssp_to_domain_hosts(4) ファイルの例 <pre># # Format: domain_name ssp_hostname # xf2 xf2-ssp ts4 ts4-ssp</pre>
関連項目	domain_create(1M), domain_remove(1M), domain_rename(1M)