

JP1 Version 9

# JP1/Cm2/Extensible SNMP Agent

解説・文法・操作書

3020-3-T04-01

## マニュアルの購入方法

このマニュアル，および関連するマニュアルをご購入の際は，  
巻末の「ソフトウェアマニュアルのサービス ご案内」をご参  
照ください。

## 対象製品

P-9D42-8A91 JP1/Cm2/Extensible SNMP Agent 09-00 (適用 OS : Solaris)

P-1M42-8A91 JP1/Cm2/Extensible SNMP Agent 09-00 (適用 OS : AIX)

P-9S42-8A91 JP1/Cm2/Extensible SNMP Agent 09-00 (適用 OS : Red Hat Enterprise Linux)

P-1J42-8A91 JP1/Cm2/Extensible SNMP Agent 09-00 (適用 OS : HP-UX (IPF))

P-9V42-8A91 JP1/Cm2/Extensible SNMP Agent 09-00 (適用 OS : Red Hat Enterprise Linux (IPF))

本ソフトウェアおよびドキュメンテーションは、ヒューレット・パッカード社からライセンスされた、ソフトウェアおよびドキュメンテーションをベースにしています。

## 輸出時の注意

本製品を輸出される場合には、外国為替および外国貿易法ならびに米国の輸出管理関連法規などの規制をご確認の上、必要な手続きをお取りください。

なお、ご不明な場合は、弊社担当営業にお問い合わせください。

## 商標類

AIX は、米国およびその他の国における International Business Machines Corporation の商標です。

AMD は、Advanced Micro Devices, Inc. の商標です。

AT は、米国およびその他の国における International Business Machines Corporation の商標です。

HP-UX は、米国 Hewlett-Packard Company のオペレーティングシステムの名称です。

Itanium は、アメリカ合衆国およびその他の国における Intel Corporation の商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Red Hat は、米国およびその他の国で Red Hat, Inc. の登録商標若しくは商標です。

RISC システム /6000 は、日本アイ・ピー・エム (株) の商品名称です。

RS/6000 は、米国およびその他の国における International Business Machines Corporation の商標です。

Solaris は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Sun は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

Sun Microsystems は、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。

UNIX は、The Open Group の米国ならびに他の国における登録商標です。

プログラムプロダクト「P-9D42-8A91」には、米国 Sun Microsystems, Inc. が著作権を有している部分が含まれています。

プログラムプロダクト「P-9D42-8A91」には、UNIX System Laboratories, Inc. が著作権を有している部分が含まれています。

## 発行

2009 年 7 月 (第 1 版) 3020-3-T04

2010 年 9 月 (第 1.1 版) 3020-3-T04-01

## 著作権

All Rights Reserved. Copyright (C) 2009, 2010, Hitachi, Ltd.

Copyright (C) 1993-1998, Hewlett-Packard Company

Copyright (C) 1989-2006, SNMP Research International, Incorporated

## 変更内容

変更内容 (3020-3-T04-01)

追加・変更内容	変更箇所
記載内容を変更した。詳細は、付録 G を参照のこと。	付録 G

# はじめに

---

このマニュアルは、次に示すプログラムプロダクトのエージェント機能および使用方法について説明したものです。

- JP1/Cm2/Extensible SNMP Agent

なお、エージェント機能を SNMP エージェントとします。

## 対象読者

このマニュアルは、SNMP エージェントを使用してネットワークを管理するネットワーク管理者、および運用や保守をするネットワーク運用者を対象にしています。また、次の内容を理解されていることを前提としています。

- UNIX ( AIX , HP-UX (IPF) , Linux(R) , Linux (IPF) ) および Solaris ) に関する知識
- SNMP プロトコルや SNMP を適用した TCP/IP ネットワークの管理手法に関する知識

## マニュアルの構成

このマニュアルは、次に示す章から構成されています。

### 第 1 章 SNMP エージェントの概要

SNMP エージェントの概要について説明しています。

### 第 2 章 SNMP エージェントの環境設定

SNMP エージェントのインストールおよび環境設定のカスタマイズについて説明しています。

### 第 3 章 SNMP エージェントの運用

SNMP エージェントの起動と停止、および SNMP エージェントの起動中の運用について説明しています。

### 第 4 章 MIB オブジェクト一覧

SNMP エージェントが実装している MIB オブジェクトの一覧と実装状況について説明しています。

### 第 5 章 コマンドとプロセス

SNMP エージェントで使用するコマンドの機能、および文法について説明します。また、プロセスの機能、および起動オプションについて説明しています。

### 第 6 章 トラブルシューティング

SNMP エージェントでトラブルの原因になりやすい項目やトラブルへの対処方法などについて説明しています。

## 関連マニュアル

このマニュアルの関連マニュアルを次に示します。必要に応じてお読みください。

- JP1 Version 8 JP1/Cm2/Network Node Manager ネットワーク管理ガイド ( 3020-3-L01 )

- JP1 Version 9 JP1/Cm2/Network Node Manager i インストールガイド ( 3020-3-T01 )
- JP1 Version 9 JP1/Cm2/Network Node Manager i セットアップガイド ( 3020-3-T02 )
- JP1 Version 9 JP1/Cm2/SNMP System Observer ( 3020-3-T11 )
- JP1 Version 7i JP1/Cm2/Internet Gateway Server ( 3020-3-G15 )
- JP1 Version 7i JP1/Cm2/ 階層管理 ( 3020-3-F98 )
- JP1 Version 6 JP1/NETM/DM Manager ( 3000-3-841 )
- JP1 Version 8 JP1/NETM/DM SubManager(UNIX(R) 用 ) ( 3020-3-L42 )
- JP1 Version 9 JP1/NETM/DM Client(UNIX(R) 用 ) ( 3020-3-S85 )

## このマニュアルでの表記

このマニュアルでは、製品名称を、略称を使って表記しています。このマニュアルでの表記と正式名称を次の表に示します。

このマニュアルでの表記		正式名称
AIX		AIX 5L V5.3 (POWER5,POWER6)
		AIX 6.1 (POWER6)
Hierarchical Agent		JP1/Cm2/Hierarchical Agent
HP-UX (IPF)		HP-UX 11i V2 (IPF)
		HP-UX 11i V3 (IPF)
JP1/Cm2/SSO		JP1/Cm2/SNMP System Observer
		JP1/Performance Management/SNMP System Observer
Linux	Red Hat Enterprise Linux	Red Hat Enterprise Linux AS4/ES4 (x86, AMD64&EM64T)
		Red Hat Enterprise Linux 5 (x86, AMD64&EM64T)
		Red Hat Enterprise Linux Advanced Platform 5 (x86, AMD64&EM64T)
Linux (IPF)	Red Hat Enterprise Linux (IPF)	Red Hat Enterprise Linux AS4 (IPF)
		Red Hat Enterprise Linux 5 (IPF)
		Red Hat Enterprise Linux Advanced Platform 5 (IPF)
NNM	HP NNM	HP Network Node Manager Software バージョン 6 以前
		HP Network Node Manager Starter Edition Software バージョン 7.5 以前
	JP1/Cm2/NNM	JP1/Cm2/Network Node Manager バージョン 7 以前
		JP1/Cm2/Network Node Manager Starter Edition 250 バージョン 8 以前
		JP1/Cm2/Network Node Manager Starter Edition Enterprise バージョン 8 以前
	NNMi	HP NNMi
JP1/Cm2/NNMi		JP1/Cm2/Network Node Manager i 09-00 以降

このマニュアルでの表記	正式名称
SNMP エージェント	JP1/Cm2/Extensible SNMP Agent
Solaris	Solaris 9 (SPARC)
	Solaris 10 (SPARC)
SubManager	JP1/Cm2/SubManager

注 Red Hat Enterprise Linux 5 および Red Hat Enterprise Linux 5 Advanced Platform 5 のバージョンは 5.1 からの対応となります。

AIX, HP-UX (IPF), Linux, Linux (IPF) および Solaris を総称して, UNIX と表記することがあります。

### このマニュアルで使用する英略語

このマニュアルで使用する英略語を, 次の表に示します。

英略語	正式名称
DLPAR	Dynamic Logical Partition
IPF	Itanium(R) Processor Family
MIB	Management Information Base
RFC	Request for Comments
SMF	Service Management Facility
SMT	Simultaneous Multi-Threading
SNMP	Simple Network Management Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol

### このマニュアルで使用する記号

このマニュアルで使用する記号を, 次の表に示します。

記号	意味
[ ]	メニュー項目, プッシュボタン, またはアイコン名称を表します。

### 定義およびコマンドの文法で使用する記号

定義およびコマンドの文法で使用する記号を, 次の表に示します。

記号	意味
[ ]	この記号で囲まれている項目は, 任意に指定できます (省略可)。 (例) [A] は「何も指定しない」か, 「A と指定する」ことを示します。
	複数の項目に対して項目間の区切りを示し, 「または」の意味を示します。 (例) 「A B C」は, 「A, B, または C」を示します。
{ }	この記号で囲まれている複数の項目の中から一つを選択することを表します。
< >	各項目の仮定値を表します。

記号	意味
《 》	省略時の仮定値を表します。
(( ))	指定できる値の範囲を表します。

## このマニュアルでのファイル名の表記

このマニュアルで使用するディレクトリ名やファイル名は、Solaris、Linux、および Linux (IPF) で動作する製品に適用します。AIX および HP-UX (IPF) で動作する製品では、ファイル名やディレクトリ名が異なるため、次の表に示す名称に置き換えてください。

Solaris、Linux、Linux (IPF) の場合	AIX の場合	HP-UX (IPF) の場合
/opt/OV/bin/snmptrap	/usr/OV/bin/snmptrap	/opt/OV/bin/snmptrap
/opt/OV/bin/systemtrap	/usr/OV/bin/systemtrap	/opt/OV/bin/systemtrap
/opt/OV/prg_samples/eagent	/usr/OV/prg_samples/eagent	/opt/OV/prg_samples/eagent
/opt/OV/newconfig /EAGENT-RUN	/usr/OV/newconfig /EAGENT-RUN	/opt/CM2/ESA/newconfig
/opt/CM2/ESA/bin/snmpstart	/usr/CM2/ESA/bin/snmpstart	/opt/CM2/ESA/bin/snmpstart
/opt/CM2/ESA/bin/snmpstop	/usr/CM2/ESA/bin/snmpstop	/opt/CM2/ESA/bin/snmpstop
/opt/CM2/ESA/bin/snmpcheck	/usr/CM2/ESA/bin/ snmpcheck	/opt/CM2/ESA/bin/snmpcheck
/opt/CM2/ESA/ext	/usr/CM2/ESA/ext	/opt/CM2/ESA/ext
/var/opt/OV/share/snmp_mibs	/usr/OV/conf/snmp_mibs	/opt/CM2/ESA/snmp_mibs
/var/opt/CM2/ESA/log/ htc_monagt1.log	/usr/CM2/ESA/log/ htc_monagt1.log	/var/opt/CM2/ESA/log/ htc_monagt1.log
/var/opt/CM2/ESA/log/ htc_monagt1.log.old	/usr/CM2/ESA/log/ htc_monagt1.log.old	/var/opt/CM2/ESA/log/ htc_monagt1.log.old
/usr/sbin/snmpdm (ただし Solaris は /opt/CM2/ESA/ bin/snmpdm)	/usr/sbin/snmpdm	/opt/CM2/ESA/bin/snmpdm
-	-	/usr/sbin/mib2agt
/usr/sbin/hp_unixagt (ただし Solaris は /opt/CM2/ESA/ bin/hp_unixagt)	/usr/sbin/hp_unixagt	/usr/sbin/hp_unixagt
/usr/sbin/trapdestagt (ただし Solaris は /opt/CM2/ESA/ bin/trapdestagt)	/usr/sbin/trapdestagt	/opt/CM2/ESA/bin/trapdestagt
/usr/sbin/extsubagt	/usr/sbin/extsubagt	/opt/CM2/ESA/bin/extsubagt
/usr/sbin/htc_unixagt1 (ただし Solaris は /opt/CM2/ESA/ bin/htc_unixagt1)	/usr/sbin/htc_unixagt1	/opt/CM2/ESA/bin/htc_unixagt1
-	-	/opt/CM2/ESA/bin/htc_unixagt2

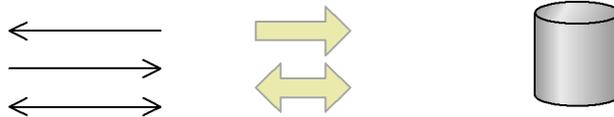
Solaris , Linux , Linux (IPF) の場合	AIX の場合	HP-UX (IPF) の場合
/usr/sbin/htc_unixagt3 (ただし Solaris は /opt/CM2/ESA/bin/htc_unixagt3)	/usr/sbin/htc_unixagt3	-
/usr/sbin/htc_monagt1 (ただし Solaris は /opt/CM2/ESA/bin/htc_monagt1)	/usr/sbin/htc_monagt1	-
/opt/CM2/ESA/bin/snmpcmdchk	/usr/CM2/ESA/bin/snmpcmdchk	-

(凡例) - : 該当しない

### 図中で使用する記号の説明

このマニュアルの図中で使用する記号を、次のように定義します。

●制御の流れ      ●データの流れ      ●ファイル



### 常用漢字以外の漢字の使用について

このマニュアルでは、常用漢字を使用することを基本としていますが、次に示す用語については、常用漢字以外の漢字を使用しています。

- 個所 (かしょ)
- 必須 (ひつす)

### KB (キロバイト) などの単位表記について

1KB (キロバイト), 1MB (メガバイト), 1GB (ギガバイト), 1TB (テラバイト), 1PB (ペタバイト) はそれぞれ  $1,024$  バイト,  $1,024^2$  バイト,  $1,024^3$  バイト,  $1,024^4$  バイト,  $1,024^5$  バイトです。



# 目次

<b>1</b>	<b>SNMP エージェントの概要</b>	<b>1</b>
1.1	SNMP エージェントとは	2
1.1.1	SNMP エージェントのシステム構成	2
1.1.2	SNMP エージェントの動作環境	3
1.2	SNMP エージェントの機能	5
1.2.1	SNMP 要求の処理	5
1.2.2	SNMP トラップの発行	6
1.2.3	拡張機能	9
1.3	SNMP エージェントのプロセス	11
1.3.1	SNMP エージェントを構成するプロセス	11
1.3.2	SNMP エージェント起動時の動作	15
1.3.3	SNMP エージェント運用時の動作	15
1.4	SNMP エージェントのコマンド	17
<b>2</b>	<b>SNMP エージェントの環境設定</b>	<b>19</b>
2.1	インストールからセットアップまでの流れ	20
2.2	インストールとアンインストール	22
2.2.1	インストール	22
2.2.2	アンインストール	26
2.2.3	バージョンアップインストール	27
2.3	インストール時の注意事項	32
2.3.1	インストール時の注意事項 (Solaris の場合)	32
2.3.2	インストール時の注意事項 (AIX の場合)	35
2.3.3	インストール時の注意事項 (Linux および Linux (IPF) の場合)	35
2.3.4	インストール時の注意事項 (HP-UX (IPF) の場合)	37
2.4	構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) のカスタマイズ	39
2.4.1	連絡先と設置場所の設定	39
2.4.2	コミュニティ名の設定	40
2.4.3	トラップのあて先の設定	46
2.4.4	構成定義ファイルの形式	47
2.5	ネイティブエージェントアダプターの設定 (HP-UX (IPF) 以外の場合)	50
2.5.1	ネイティブエージェントアダプターの機能	50
2.5.2	ネイティブエージェントの設定 (AIX, Solaris の場合)	53

2.5.3	ネイティブエージェントアダプターの設定方法	53
2.5.4	ネイティブエージェントアダプターの使用上の注意事項	57
2.6	拡張 MIB オブジェクトの定義	59
2.6.1	MIB オブジェクトの定義	61
2.6.2	拡張 MIB 定義ファイルの作成	62
2.6.3	SNMP 要求時に実行するコマンドの作成	71
2.6.4	SNMP 要求時に処理するファイルの作成	75
2.6.5	サブエージェントの再構成	76
2.6.6	マネージャーのコマンドによるオブジェクトの確認	77
2.6.7	すべての SNMP エージェントの設定	77
2.6.8	拡張 MIB オブジェクトのマネージャーへのコピー	77
2.6.9	マネージャーへの MIB の統合	77
2.6.10	複数の拡張 MIB 定義ファイルの設定	78
2.6.11	拡張 MIB オブジェクトの定義例	82
2.7	拡張トラップの定義	99
2.7.1	拡張トラップの定義方法	99
2.7.2	拡張トラップの利用方法	99
2.7.3	スクリプト例	100
2.8	クラスタ環境で運用する場合の設定	102
2.9	セットアップ時の注意事項	105
2.9.1	セットアップ時の注意事項 (Linux および Linux (IPF) の場合)	106

## 3

SNMP エージェントの運用	107	
3.1	SNMP エージェントの起動と停止	108
3.1.1	SNMP エージェントの起動	108
3.1.2	SNMP エージェントの停止	112
3.1.3	ネイティブエージェントの起動と停止	113
3.2	SNMP エージェントの SNMP 受信ポートの変更	114
3.2.1	SNMP エージェントの SNMP 受信ポートの変更方法	114
3.2.2	ネイティブエージェント snmpd の SNMP 受信ポートの変更 (AIX の場合)	115
3.3	最大接続サブエージェント数の変更	117
3.4	バックアップとリストア	118
3.4.1	設定ファイルのバックアップとリストア	118
3.4.2	フルバックアップ・フルリストア時の注意事項	118
3.5	運用上の注意事項	120
3.5.1	運用上の注意事項 (Solaris の場合)	123

3.5.2	運用上の注意事項（AIX の場合）	123
-------	-------------------	-----

## 4

	<b>MIB オブジェクト一覧</b>	<b>129</b>
4.1	標準 MIB オブジェクト一覧	130
4.1.1	標準 MIB オブジェクトの構成	130
4.1.2	標準 MIB オブジェクトの内容	131
4.1.3	標準 MIB オブジェクトの実装状況	139
4.2	HP 企業固有 MIB オブジェクト一覧	141
4.2.1	HP 企業固有 MIB オブジェクトの構成	141
4.2.2	HP 企業固有 MIB オブジェクトの内容	142
4.2.3	HP 企業固有 MIB オブジェクトの実装状況	152
4.3	日立企業固有 MIB オブジェクト一覧	160
4.3.1	日立企業固有 MIB オブジェクトの構成	160
4.3.2	日立企業固有 MIB オブジェクトの内容	162
4.3.3	日立企業固有 MIB オブジェクトの実装状況	199

## 5

	<b>コマンドとプロセス</b>	<b>231</b>
	コマンド	232
	コマンドの詳細	233
	jp1esalog.sh.def	234
	snmpcheck	238
	snmpcmdchk	239
	snmpstart	240
	snmpstop	241
	snmptrap	242
	systemtrap	245
	プロセス	246
	プロセスの詳細	247
	snmpdm	248
	extsubagt	252
	hp_unixagt	255
	htc_monagt1	257
	htc_unixagt1	260
	htc_unixagt2	263
	htc_unixagt3	265
	naaagt	267

trapdestagt	269
-------------	-----

## 6

トラブルシューティング	271
6.1 対処の手順	272
6.2 問題の把握	273
6.3 ログの取得	274
6.4 資料の採取	278
6.4.1 マスターエージェントの送受信パケットダンプの取得	278
6.4.2 ネイティブエージェントアダプターの送受信パケットダンプの取得	279
6.4.3 マスターエージェントの詳細トレースの取得	280
6.4.4 コミュニティ名不正時のログ取得	282
6.5 トラブルへの対処方法	283
6.5.1 SNMP エージェント起動時の問題	283
6.5.2 SNMP エージェントの運用上の問題	283
6.6 ログ情報の採取方法	287

## 付録

付録 A SNMP エージェントのファイルの一覧	289
付録 A.1 SNMP エージェントのファイルの一覧 (Solaris の場合)	290
付録 A.2 SNMP エージェントのファイルの一覧 (AIX の場合)	294
付録 A.3 SNMP エージェントのファイルの一覧 (Linux の場合)	299
付録 A.4 SNMP エージェントのファイルの一覧 (HP-UX (IPF) の場合)	302
付録 A.5 SNMP エージェントのファイルの一覧 (Linux (IPF) の場合)	305
付録 B ポート番号一覧	310
付録 B.1 SNMP エージェントで使用するポート番号	310
付録 B.2 ファイアウォールの通過方向	310
付録 C カーネルパラメーター一覧	312
付録 D SNMP エージェントの前提とするパッチ, プロセス (サービス) 一覧	313
付録 E IBM RISC システム /6000 SP2(RS/6000 SP2) へのインストールとアンインストール	314
付録 E.1 IBM RISC システム /6000 SP2(RS/6000 SP2) へのインストール	314
付録 E.2 IBM RISC システム /6000 SP2(RS/6000 SP2) からのアンインストール	317
付録 F 各バージョンの変更内容	318
付録 G (3020-3-T04-01) の変更内容	320
付録 H 用語解説	331

# 索引



# 1

## SNMP エージェントの概要

SNMP エージェントは、マネージャーシステム、サブマネージャーシステム、およびエージェントシステムで動作し、それぞれのシステムを管理します。この章では、SNMP エージェントの概要について説明します。

- 
- 1.1 SNMP エージェントとは
  - 1.2 SNMP エージェントの機能
  - 1.3 SNMP エージェントのプロセス
  - 1.4 SNMP エージェントのコマンド
-

## 1.1 SNMP エージェントとは

---

SNMP エージェントは、TCP/IP ネットワークを管理するためのエージェントです。SNMP エージェントは、インターネットのネットワーク管理プロトコルである SNMP を使用して、マネージャーとの間で MIB と呼ばれる管理情報を交換し、TCP/IP ネットワークを管理します。

なお、このマニュアルでは SNMP エージェントの種別を次のように定義しています。

- SNMP エージェント  
JP1/Cm2/Extensible SNMP Agent
- ネイティブエージェント  
OS が提供する SNMP エージェント
- 他社製品 SNMP エージェント  
他社が提供する SNMP エージェント

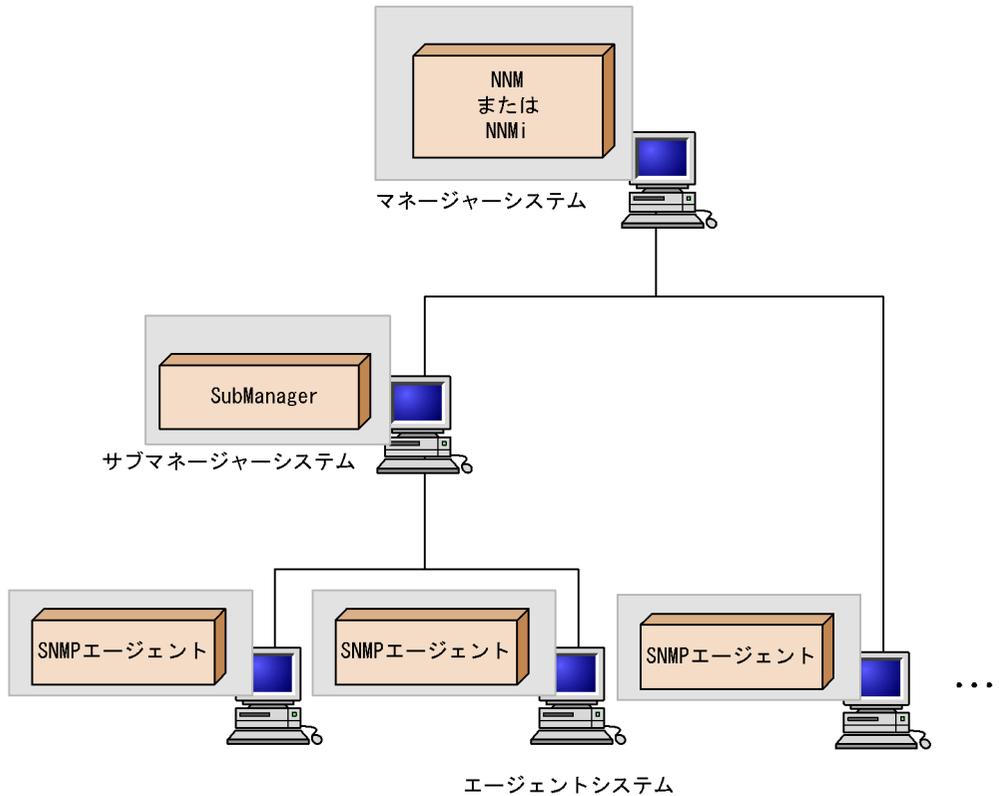
### 1.1.1 SNMP エージェントのシステム構成

SNMP エージェントは、ネットワークを構成するマネージャーシステム、サブマネージャーシステム、およびエージェントシステムで動作します。

このマニュアルでは、NNM または NNMi がインストールされているマシンをマネージャーシステム、SubManager がインストールされているマシンをサブマネージャーシステム、SNMP エージェントがインストールされているマシンをエージェントシステムと呼びます。

SNMP エージェントのシステム構成例を次の図に示します。

図 1-1 SNMP エージェントのシステム構成例



注 NNMまたはNNMiはSNMPエージェントと同じシステムに存在していることもあります。SubManagerとSNMPエージェントは同じシステムでは動作できません。

## 1.1.2 SNMP エージェントの動作環境

SNMP エージェントが動作するシステムと適用 OS を次の表に示します。

表 1-1 SNMP エージェントが動作するシステムと適用 OS

製品名	稼働システム	適用 OS
Extensible SNMP Agent	日立 HA8500 シリーズ, HP Integrity サーバおよびその互換機, BladeSymphony	HP-UX (IPF)
	SUN SPARC シリーズ, SUN Fire シリーズ, SUN Netra シリーズ, SUN Ultra シリーズ, およびこれらの互換機, SUN Blade シリーズ, PRIMEPOWER	Solaris
	EP8000, IBM pSeries, IBM RISC システム /6000 およびその互換機, IBM p5	AIX

## 1. SNMP エージェントの概要

製品名	稼働システム	適用 OS
	Linux を搭載できる PC/AT 互換機, BladeSymphony, HA8000	Linux Linux (IPF)

### 注

SNMP エージェントは 64bit カーネルモードの OS 上で動作できます。ただし、64bit カーネルモードの OS 上で動作させた場合でも、SNMP エージェント自体は 32bit アプリケーションとして動作します。

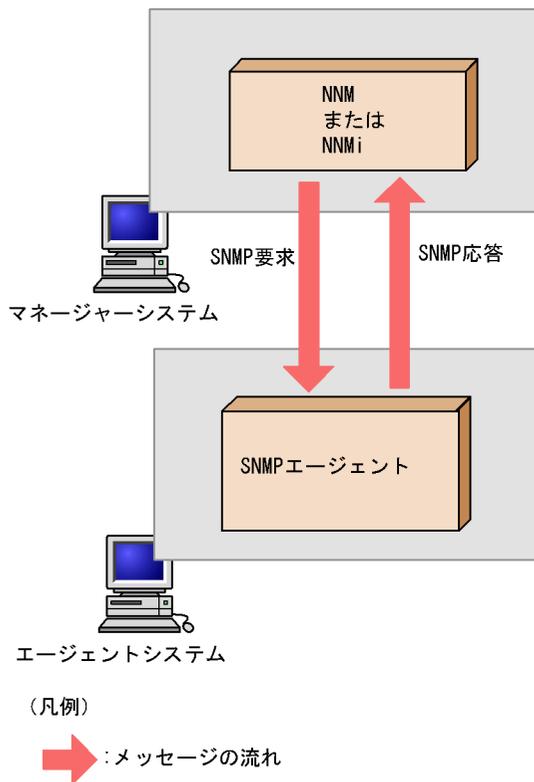
## 1.2 SNMP エージェントの機能

SNMP エージェントには、マネージャーからの SNMP 要求に応答する機能とエージェントシステムで発生する変化（SNMP トラップ）をマネージャーに送信する機能があります。また、拡張機能として、ユーザー独自の MIB オブジェクト（拡張 MIB オブジェクト）やユーザー独自の SNMP トラップ（拡張トラップ）の定義ができます。

### 1.2.1 SNMP 要求の処理

SNMP 要求は、SNMP エージェントが管理する MIB の値に対するマネージャーからのアクセス要求です。SNMP 要求の処理の概要を次の図に示します。

図 1-2 SNMP 要求の処理の概要



SNMP 要求の種類には、SNMP GET（取得）要求、SNMP SET（設定）要求、SNMP GET NEXT（取得）要求があります。SNMP エージェントは、受信した SNMP 要求を解析し、指定された MIB 値の取得や設定の処理を実行します。そのあと、処理結果を含む応答メッセージを作成し、マネージャーに応答します。

なお、SNMP 要求の処理は SNMPv1 および SNMPv2c に対応します。

## 1. SNMP エージェントの概要

SNMP エージェントが取得および設定できる MIB オブジェクトを次に示します。

### (1) SNMP エージェントが管理する MIB オブジェクト

SNMP エージェントが管理する MIB オブジェクトとして、標準 MIB-II オブジェクトに加え、HP 企業固有 MIB オブジェクトと日立企業固有 MIB オブジェクトを実装しています。SNMP エージェントの MIB の実装状況については、「4. MIB オブジェクト一覧」を参照してください。

### (2) ネイティブエージェントが提供する MIB オブジェクト

ネイティブエージェントとは、システムのベンダーなどによってシステム標準であらかじめ提供されているエージェントのことです。

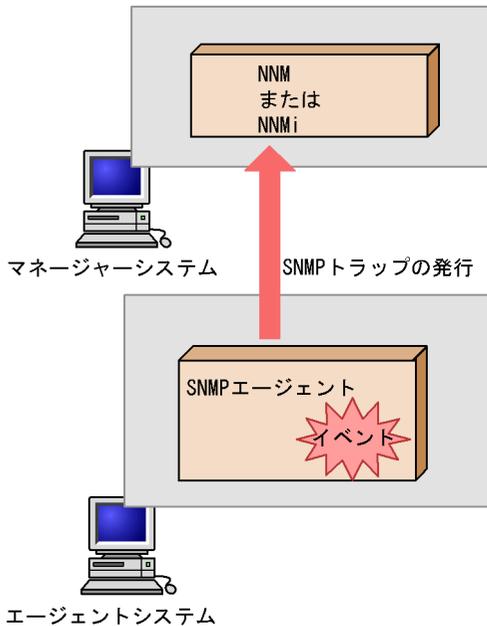
HP-UX (IPF) では、ネイティブエージェントと SNMP エージェントのマスターエージェントが直接通信して、ネイティブエージェントが提供する MIB オブジェクトを取得します。

HP-UX (IPF) 以外の OS では、SNMP エージェントのネイティブエージェントアダプター機能を使用して、ネイティブエージェントが提供する MIB オブジェクトも SNMP エージェントを通じて取得できます。ネイティブエージェントアダプターの機能およびネイティブエージェントアダプターの設定については、「2.5 ネイティブエージェントアダプターの設定 (HP-UX (IPF) 以外の場合)」を参照してください。

## 1.2.2 SNMP トラップの発行

SNMP エージェントはエージェントシステムで発生した変化を SNMP トラップとしてマネージャーに送信します。SNMP トラップの発行の概要を次の図に示します。

図 1-3 SNMP トラップの発行の概要



(凡例)

➡ : メッセージの流れ

SNMP エージェントは、起動時やマネージャからの不当な SNMP 要求を受信した場合などに、SNMP トラップメッセージを作成し、マネージャに送信します。なお、SNMP トラップメッセージは SNMPv1 に対応しています。

#### 参考

ユーザーはマネージャでどの SNMP トラップを重要とみなし、アラームとして表示するかを制御できます。マネージャで表示されたアラームを簡単にモニターし、ネットワークを正常に保つために適切なアクションをとることができます。

次に、SNMP トラップについて説明します。

### (1) SNMP トラップの標準トラップ番号

SNMP トラップには一般トラップと拡張トラップがあります。それぞれの SNMP トラップは RFC1157 で次のように規定されています。

#### 一般トラップ

RFC1157 で規定されている標準トラップ番号 0 ~ 5 で識別されるトラップ。

#### 拡張トラップ

## 1. SNMP エージェントの概要

RFC1157 で規定されている標準トラップ番号 6 とユーザーが任意に指定した固有のトラップ番号を結合した番号で識別されるトラップ。

SNMP トラップの種別と標準トラップ番号を次の表に示します。

表 1-2 SNMP トラップの種別と標準トラップ番号一覧

SNMP トラップの種別	標準トラップ番号	意味
一般トラップ	0	ColdStart
	1	WarmStart
	2	LinkDown
	3	LinkUp
	4	AuthenticationFailure
	5	EgpNeighborLoss
拡張トラップ	6	EnterpriseSpecific

注 SNMP エージェントが発行する一般トラップは、ColdStart トラップおよび、AuthenticationFailure トラップです。その他の一般トラップは発行しません。

ここでは、一般トラップについて説明します。拡張トラップについては、「1.2.3(2) 拡張トラップの定義」を参照してください。

### (2) SNMP トラップ発行時のエージェントアドレス

SNMP トラップ発行時のエージェントアドレスは、SNMP エージェントがインストールされたマシンの IP アドレスです。該当するシステムの OS の関数を使用して取得したホスト名を、OS の関数を使用して IP アドレスに変換して取得します。

### (3) エンタープライズ ID

SNMP トラップ中に設定されるエンタープライズ ID を次の表に示します。

表 1-3 トラップ中に設定するエンタープライズ ID

SNMP エージェントが稼働している OS	エンタープライズ ID	該当するシステムの構成
Solaris	.1.3.6.1.4.1.116.3.8.1.1	SNMP エージェントおよび NNM がシステムに共存している。
	.1.3.6.1.4.1.116.3.8.1.2	SubManager
	.1.3.6.1.4.1.116.3.8.1.3	SNMP エージェントおよび NNM がシステムに共存していない。
	.1.3.6.1.4.1.116.3.8.1.4	SNMP エージェント、NNM および Hierarchical Agent がシステムに共存している。

SNMP エージェントが稼働している OS	エンタープライズ ID	該当するシステムの構成
HP-UX (IPF)	.1.3.6.1.4.1.116.3.9.1.1	SNMP エージェントおよび NNM がシステムに共存している。
	.1.3.6.1.4.1.116.3.9.1.2	SubManager
	.1.3.6.1.4.1.116.3.9.1.3	SNMP エージェントおよび NNM がシステムに共存していない。
	.1.3.6.1.4.1.116.3.9.1.4	SNMP エージェント、NNM および Hierarchical Agent がシステムに共存している。
AIX	.1.3.6.1.4.1.116.3.13.1.2	SubManager
	.1.3.6.1.4.1.116.3.13.1.3	SNMP エージェントがインストールされている。
Linux , Linux (IPF)	.1.3.6.1.4.1.116.3.14.1.3	SNMP エージェントがインストールされている。

注 SNMP エージェントと NNMi がシステムに共存している場合でも末尾が 3 のエンタープライズ ID が設定されます。

SNMP トラップ中に設定するエンタープライズ ID は、SNMP エージェントの標準 MIB オブジェクトである System グループの sysObjectID の値が設定されます。

#### 参考

マネージャーで SNMP トラップに対してイベント設定が必要な場合は、SNMP エージェントが発行する SNMP トラップ中に設定されているエンタープライズ ID を設定してください。なお、NNM のイベント設定には、SNMP エージェントから通知される SNMP トラップ情報が設定されています。

## 1.2.3 拡張機能

SNMP エージェントの次に示す拡張機能について説明します。

- 拡張 MIB オブジェクトの定義
- 拡張トラップの定義

### (1) 拡張 MIB オブジェクトの定義

ハードウェアベンダーや標準化組織で定義される MIB を、拡張 MIB オブジェクトとして実装できます。ASN.1 記述に従って、指定したファイルに拡張したい MIB オブジェクトを定義します。SNMP エージェントがこの MIB オブジェクトに対して SNMP 要求を受け取ったときに実行するアクションもあわせて定義します。これによって SNMP エージェントは、マネージャーからの SNMP 要求を受け取ったとき、対応するアクションを

## 1. SNMP エージェントの概要

実行し、その処理結果をマネージャーに応答します。

拡張 MIB オブジェクトを使用して、次のことができます。

- ユーザーの独自情報を MIB オブジェクトとして管理できます。
- ユーザー独自のアプリケーションを起動、または停止できます。

### (2) 拡張トラップの定義

SNMP エージェントでは、ユーザー独自のトラップを拡張トラップとしてマネージャーに通知できます。拡張トラップのトラップ番号は、トラップ番号 6 と固有のトラップ番号で定義します。トラップ番号については、「1.2.2(1) SNMP トラップの標準トラップ番号」を参照してください。

SNMP エージェントでは、通知のために `snmptrap` コマンドを用意しています。例えば、重要なプロセスが停止した場合に、`snmptrap` コマンドを使用してマネージャーに拡張トラップとして通知すれば、そのあとマネージャーが SNMP SET 要求を使用してプロセスを再起動できます。`snmptrap` コマンドの詳細については、「5. コマンドとプロセス」の「`snmptrap`」を参照してください。

## 1.3 SNMP エージェントのプロセス

---

ここでは、SNMP エージェントを構成するプロセスについて説明します。また、SNMP エージェントの起動時および運用時のプロセスの動作についても説明します。

### 1.3.1 SNMP エージェントを構成するプロセス

SNMP エージェントはマスターエージェントとサブエージェントで構成されます。また、HP-UX (IPF) 以外の OS で稼働する SNMP エージェントでは定期的に OS からの情報を取得する情報収集デーモンも動作します。

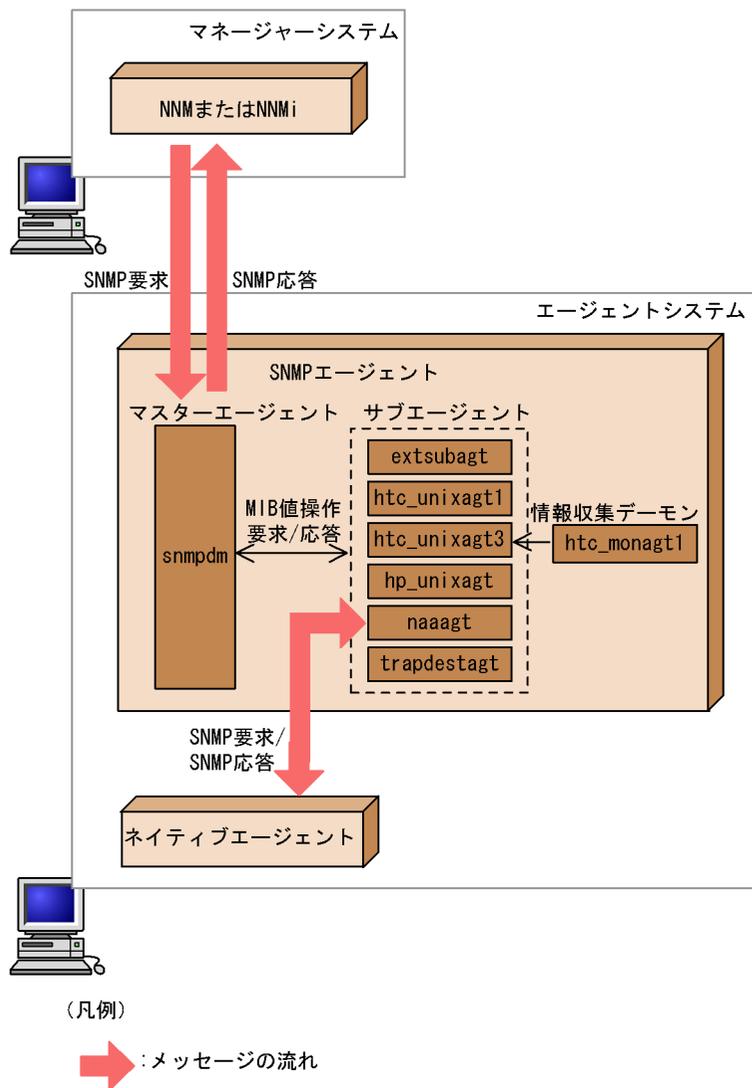
システムの OS によって、SNMP エージェントを構成するプロセスは異なります。適用 OS ごとに SNMP エージェントのプロセス構成を次に示します。

Solaris , AIX , Linux および Linux (IPF) の場合

Solaris , AIX , Linux および Linux (IPF) の場合の SNMP エージェントのプロセス構成を次に示します。

# 1. SNMP エージェントの概要

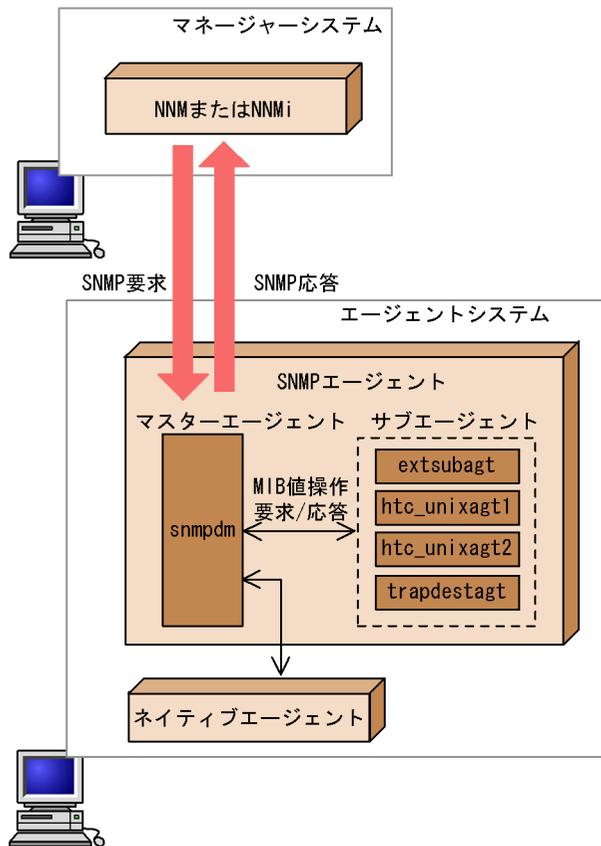
図 1-4 SNMP エージェントのプロセス構成 ( Solaris , AIX , Linux および Linux (IPF) の場合 )



## HP-UX (IPF) の場合

HP-UX (IPF) の場合の SNMP エージェントのプロセス構成を次に示します。

図 1-5 SNMP エージェントのプロセス構成 (HP-UX (IPF) の場合)



(凡例)

➡ :メッセージの流れ

### (1) マスターエージェントで動作するプロセス

マスターエージェントで動作するプロセスを次に示します。

- snmpdm  
マネージャからの SNMP 要求を受信します。その受信したメッセージをサブエージェントに通知します。また、サブエージェントからの応答をマネージャに送信します。

### (2) サブエージェントで動作するプロセス

サブエージェントで動作するプロセスを次に示します。

- extsubagt

## 1. SNMP エージェントの概要

拡張 MIB オブジェクトを提供します。

- htc\_unixagt1  
日立企業固有 MIB を提供します。提供する MIB オブジェクトについては、「5. コマンドとプロセス」の「htc\_unixagt1」を参照してください。
- htc\_unixagt2  
日立企業固有 MIB を提供します。提供する MIB オブジェクトについては、「5. コマンドとプロセス」の「htc\_unixagt2」を参照してください。
- htc\_unixagt3  
日立企業固有 MIB を提供します。提供する MIB オブジェクトについては、「5. コマンドとプロセス」の「htc\_unixagt3」を参照してください。
- hp\_unixagt  
HP 企業固有 MIB オブジェクトを提供します。提供する MIB オブジェクトについては、「5. コマンドとプロセス」の「hp\_unixagt」を参照してください。
- naaagt  
ネイティブエージェントアダプター機能を提供します。
- trapdestagt  
HP 企業固有 MIB オブジェクトの Trap グループ ( hp.nm.snmp.trapMIB ) を提供します。

サブエージェントのプロセスは OS によって異なります。SNMP エージェントが提供するサブエージェントのプロセスと適用 OS の一覧を次の表に示します。

表 1-4 SNMP エージェントが提供するサブエージェントのプロセスと適用 OS

SNMP エージェントが提供するサブエージェントのプロセス	適用 OS			
	HP-UX (IPF)	Solaris	AIX	Linux , Linux (IPF)
extsubagt <sup>1</sup>				
htc_unixagt1				
htc_unixagt2		x	x	x
htc_unixagt3	x			
hp_unixagt	x			
naaagt	x			2
trapdestagt				

(凡例)

- : 提供する。
- x : 提供しない。

注 1

extsubagt プロセスは、ユーザーが拡張 MIB 定義ファイルを設定した時に動作します。インストール直後には拡張 MIB 定義ファイルは設定されていません。必要に応じて拡張 MIB 定義

ファイルを設定してください。拡張 MIB 定義ファイルの設定方法については、「2.6.2 拡張 MIB 定義ファイルの作成」を、複数の拡張 MIB 定義ファイルの設定方法については、「2.6.10 複数の拡張 MIB 定義ファイルの設定」を参照してください。

注 2

ネイティブエージェントが起動している場合に起動します。

### (3) 情報収集デーモンプロセス

情報収集デーモンとして動作するプロセスを次に示します。

- htc\_monagt1  
CPU の利用に関する情報を提供します。

htc\_monagt1 プロセスの提供は OS によって異なります。htc\_monagt1 プロセスの適用 OS の一覧を次の表に示します。

表 1-5 htc\_monagt1 プロセスの適用 OS

情報収集デーモン	適用 OS			
	HP-UX (IPF)	Solaris	AIX	Linux , Linux (IPF)
htc_monagt1	x			

(凡例)

- : 提供する。
- x : 提供しない。

## 1.3.2 SNMP エージェント起動時の動作

SNMP エージェントのプロセスは、通常、システムの起動とともに自動的に起動され、次のように動作します。

- マスターエージェントは起動時に構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) を読み込みます。
- サブエージェントは起動時に自分自身が保持する MIB をマスターエージェントに登録します。

SNMP エージェントが提供するプロセスはそれぞれ起動オプションおよび環境変数を定義するファイルを持っていて、プロセスの動作をカスタマイズすることができます。プロセスの起動オプションは、コマンド形式で指定できます。起動オプションの詳細については、「5. コマンドとプロセス」の「プロセス」を参照してください。

## 1.3.3 SNMP エージェント運用時の動作

マネージャーからの SNMP 要求は、すべてマスターエージェントが受信します。マス

## 1. SNMP エージェントの概要

ターエージェントは SNMP 要求中にサブエージェントが登録した MIB が含まれていれば、そのサブエージェントに対して MIB 値操作要求を送信します。サブエージェントは指定された MIB 値操作の実行結果をマスターエージェントに MIB 値操作応答を返信します。マスターエージェントは、この返信結果をマネージャーに SNMP 応答します。

マスターエージェント、およびサブエージェントで発生したエラーは `snmpd.logn` (n : ログファイルの面番号) ファイルに記録されます。情報収集デーモンで発生したエラーは `htc_monagt1.log` ファイルに出力されます。また、ログファイルに出力されるログの取得内容の種類を特定することもできます。ログの取得については、「6.3 ログの取得」を参照してください。

## 1.4 SNMP エージェントのコマンド

SNMP エージェントは次のコマンドを提供しています。

表 1-6 SNMP エージェントが提供するコマンド一覧

分類	コマンド名	機能
SNMP エージェントの起動・停止	snmpstart	SNMP エージェントを起動します。
	snmpstop	SNMP エージェントを停止します。
トラップの発行	snmptrap	SNMP トラップを発行します。
	systemtrap	システムトラップを発行します。
状態のリスト表示	snmpcheck	マスターエージェント、およびサブエージェントの起動、停止状態を表示します。
	snmpcmdchk	SNMP エージェントが MIB 値を作成するために必要とする OS コマンドのインストール状況を表示します。
障害情報の採取	jp1esalog.sh.def	障害発生時に該当するマシンのシステム情報を採取します。

注 SNMP エージェントでは、上記のコマンド以外に、SNMP エージェントのプロセスの制御をコマンド形式で提供しています。プロセスの詳細については、「5. コマンドとプロセス」を参照してください。



# 2

## SNMP エージェントの環境設定

この章では、SNMP エージェントのインストールおよび環境設定のカスタマイズについて説明しています。

- 
- 2.1 インストールからセットアップまでの流れ

---

  - 2.2 インストールとアンインストール

---

  - 2.3 インストール時の注意事項

---

  - 2.4 構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) のカスタマイズ

---

  - 2.5 ネイティブエージェントアダプターの設定 ( HP-UX (IPF) 以外の場合 )

---

  - 2.6 拡張 MIB オブジェクトの定義

---

  - 2.7 拡張トラップの定義

---

  - 2.8 クラスタ環境で運用する場合の設定

---

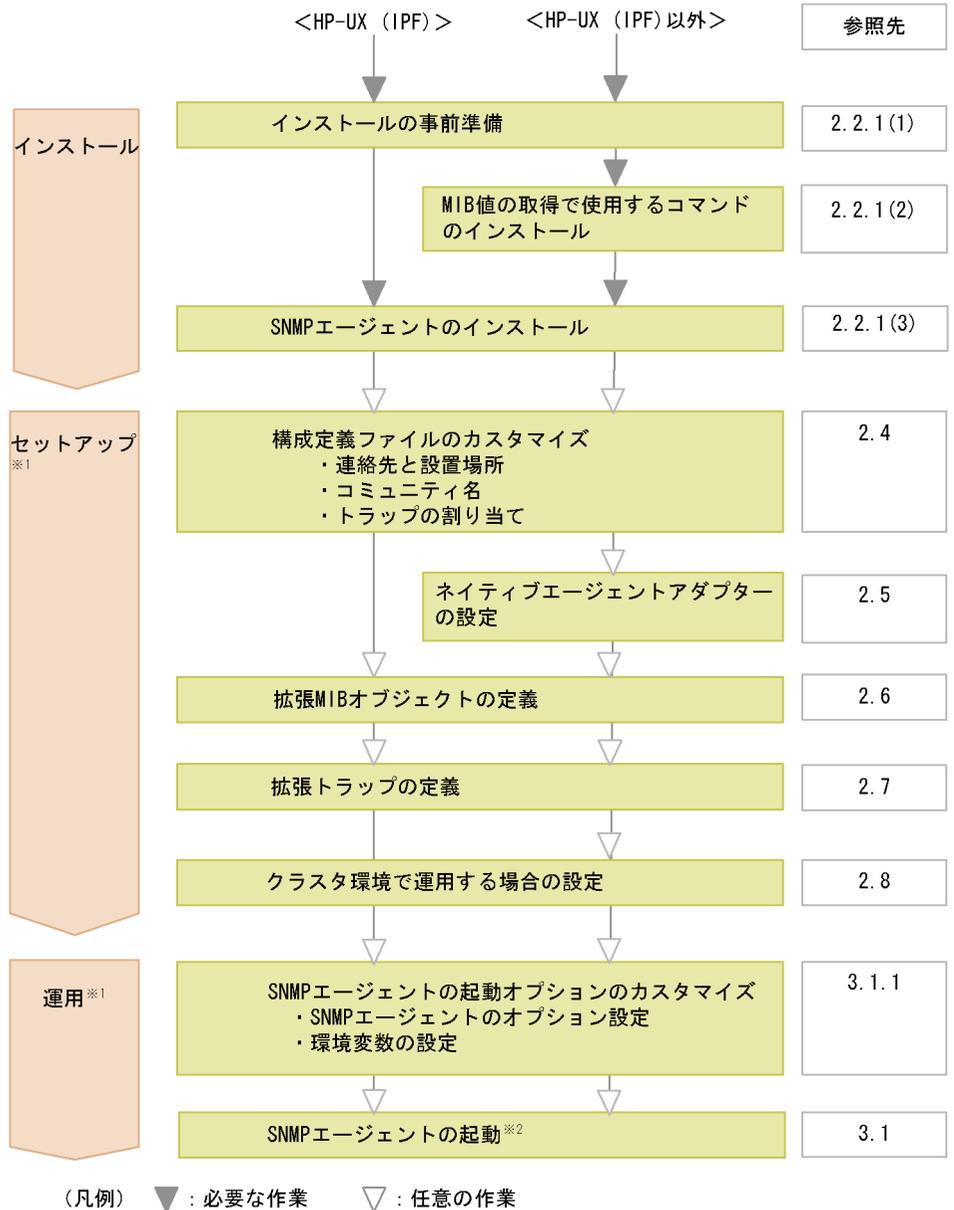
  - 2.9 セットアップ時の注意事項
-

## 2.1 インストールからセットアップまでの流れ

---

SNMP エージェントのインストールからセットアップまでの流れを次の図に示します。

図 2-1 SNMP エージェントのインストールからセットアップまでの流れ



注※1 SNMPエージェントはインストール後にデフォルトで動作します。セットアップ以降はご使用の環境に合わせてカスタマイズしてください。

注※2 SNMPエージェントはシステムの起動とともに自動的に起動されます。必要に応じて手でSNMPエージェントを起動してください。

## 2.2 インストールとアンインストール

---

この節では、SNMP エージェントのインストールおよびアンインストールの手順について説明します。バージョンアップインストールについては、「2.2.3 バージョンアップインストール」を参照してください。

### 2.2.1 インストール

ここでは、インストールの事前準備、および次のインストール手順について説明します。

- MIB 値の取得で使用するコマンドのインストール (HP-UX (IPF) 以外の場合)
- SNMP エージェントのインストール

#### (1) インストールの事前準備

SNMP エージェントをインストールする前の準備作業を次に示します。

SNMP エージェントの前提パッチがインストールされていることを確認してください。

前提パッチがインストールされていない場合、不正な MIB 値を応答したり、MIB 値が取得できなかつたりするなどの問題が発生します。SNMP エージェントの前提パッチの詳細については「付録 D SNMP エージェントの前提とするパッチ、プロセス (サービス) 一覧」を参照してください。

SNMP エージェントの実行に必要なリソースを割り当てるために、OS のカーネルパラメーターの設定値を調整してください。

カーネルパラメーターの詳細については、「付録 C カーネルパラメーター一覧」を参照してください。

ネイティブエージェントが構築されていることを確認してください。

SNMP エージェントには前提プログラムとしてネイティブエージェントが必要です。ネイティブエージェントのインストールについては、「2.3.1 インストール時の注意事項 (Solaris の場合)」, または「2.3.3 インストール時の注意事項 (Linux および Linux (IPF) の場合)」を参照してください。

#### (2) MIB 値の取得で使用するコマンドのインストール (HP-UX (IPF) 以外の場合)

SNMP エージェントは提供する MIB 値の一部を取得するために、OS のコマンドを使用しています。SNMP エージェントが使用する OS のコマンドについては、「(a) MIB 値の取得で使用するコマンド」を参照してください。

SNMP エージェントをインストールする前に、これらのコマンドをインストールしてください。また、SNMP エージェントのインストール後に、これらのコマンドが対象マシンにインストールされていることを確認してください。これらのコマンドがインストー

ルされていない場合、SNMP エージェントは MIB 値を取得できなくなるか、または不正な MIB 値を応答します。

インストールされているかどうかのチェックは `snmpcmdchk` コマンドを使用します。

(例)

Solaris の場合の実行例を次に示します。

```
#/opt/CM2/ESA/bin/snmpcmdchk
/etc/prtconf          installed.
/usr/bin/sar          installed.
/etc/swap             installed.
/usr/bin/pagesize     installed.
/usr/bin/mpstat       Not installed.
```

(凡例) installed : インストール済。

Not installed : 未インストール。

(a) MIB 値の取得で使用するコマンド

SNMP エージェントが MIB 値の取得で使用するコマンドを、OS 別に次の表に示します。

表 2-1 SNMP エージェントが MIB 値の取得で使用するコマンド

適用 OS	MIB 値の取得で使用するコマンド	コマンドの使用方法
Solaris	/etc/prtconf	prtconf
	/usr/bin/sar	sar 5 1 sar -r 5 sar -d 5 sar -d 300 1
	/etc/swap	swap -s
	/usr/bin/pagesize	pagesize
	/usr/bin/mpstat	mpstat 300 2
AIX <sup>1</sup>	/usr/bin/iostat	iostat -d
	/usr/sbin/lsdev	lsdev -Cc memory
	/usr/sbin/lsattr	lsattr -E
	/usr/sbin/lspcs	lspcs -a
	/usr/bin/ps	ps -e ps ug
	/usr/bin/uptime	uptime
	/usr/bin/vmstat	vmstat -f vmstat -s

## 2. SNMP エージェントの環境設定

適用 OS	MIB 値の取得で使用するコマンド	コマンドの使用方法
	/usr/sbin/sar	sar -P ALL 300 1 sar -d 300 1
	/usr/bin/svmon	svmon -G
Linux , Linux (IPF) <sup>2</sup>	/usr/bin/vmstat	vmstat
	/bin/ps	ps -e
	/usr/bin/uptime	uptime
	/usr/bin/free	free
	/usr/bin/mpstat	mpstat -P ALL 300 1 mpstat 300 1

注 1 AIX 5L で SNMP エージェントを使用するときは、OS で標準提供しているファイルセット perfagent.tools に含まれる svmon コマンドをインストールしてください。また、AIX 6.1 の場合は、ファイルセット bos.perf.tools に含まれる svmon コマンドをインストールしてください。

注 2 Linux および Linux (IPF) で SNMP エージェントを使用するときは、mpstat コマンドは sysstat パッケージに入っています。

### (3) SNMP エージェントのインストール

SNMP エージェントをインストールするには、Hitachi PP Installer を使用する方法と、JP1/NETM/DM を使用する方法があります。JP1/NETM/DM を使用したりリモートインストールに失敗した場合は、Hitachi PP Installer でインストールをし直してください。なお、JP1/NETM/DM を使ったりリモートインストールについては、マニュアル「JP1/NETM/DM Manager」を参照してください。

ここでは、Hitachi PP Installer を使用する方法について説明します。

SNMP エージェントのインストールは、提供媒体に格納されている Hitachi PP Installer を使用します。

次の手順で SNMP エージェントをインストールします。

1. SNMP エージェントをインストールするマシンに root 権限でログインする。  
Hitachi PP Installer を使用する前に、root 権限でシステムにログインするか、または su コマンドでユーザーの権限を root に変更してください。
2. すべてのプログラムを終了する。  
JP1 シリーズのほかのプログラムが動作している場合は、必ず停止しておいてください。
3. Hitachi PP Installer を実行する。  
Hitachi PP Installer の指示に従って SNMP エージェントをインストールしてください。Hitachi PP Installer の操作手順については、「(a) Hitachi PP Installer の使用方法」を参照してください。

### (a) Hitachi PP Installer の使用方法

Hitachi PP Installer は、SNMP エージェントの提供媒体に格納されています。ここでは Hitachi PP Installer を起動して、SNMP エージェントをインストールする手順について説明します。なお、Hitachi PP Installer の詳細については、提供媒体のリリースノートを参照してください。

#### Hitachi PP Installer の起動

Hitachi PP Installer の起動手順を次に示します。

1. ドライブに、SNMP エージェントの提供媒体 CD-ROM をセットする。
2. CD-ROM をマウントする。  
マウント方法は、ご使用の OS、ハードウェアおよび環境によって異なります。マウント方法の詳細は OS のマニュアルを参照してください。
3. Hitachi PP Installer をインストールおよび起動する。  
CD-ROM のディレクトリ名やファイル名は、マシンの環境によっては記載されている内容と見え方が異なる場合があります。ls コマンドで確認の上、表示されたファイル名を入力してください。
4. CD-ROM をアンマウントする。  
インストール完了後、アンマウントします。アンマウント方法の詳細は OS のマニュアルを参照してください。

#### SNMP エージェントのインストール方法

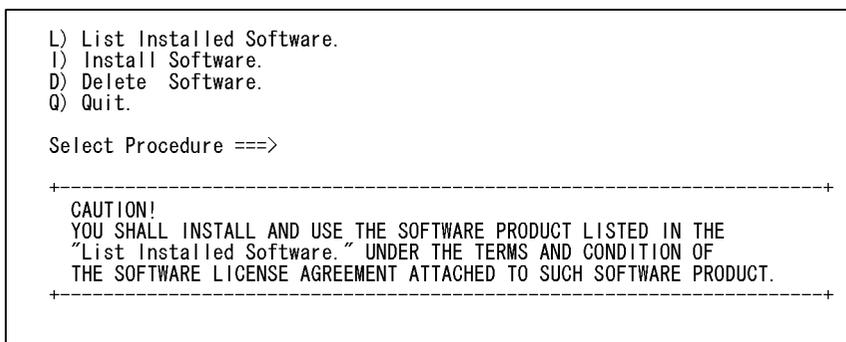
Hitachi PP Installer を使用して SNMP エージェントをインストールする方法を説明します。

次のコマンドを実行して、Hitachi PP Installer を起動します。

```
/etc/hitachi_setup
```

Hitachi PP Installer を起動すると、初期画面が表示されます。

図 2-2 Hitachi PP Installer の初期画面例



初期画面で「I」を入力すると、インストールできるプログラムの一覧が表示されます。インストールするプログラムにカーソルを移動し、スペースバーで選択します。さらに「I」を入力すると、SNMP エージェントがインストールされます。インストール完了後、「Q」を入力すると初期画面に戻ります。

#### SNMP エージェントの削除

次のコマンドを実行して、Hitachi PP Installer を起動します。

```
/etc/hitachi_setup
```

Hitachi PP Installer の初期画面が表示されます。初期画面については、図 2-2 を参照してください。

初期画面で「D」を入力すると、削除できるソフトウェアの一覧が表示されます。削除したいソフトウェアにカーソルを移動し、スペースバーで選択します。さらに「D」を入力すると、ソフトウェアが削除されます。削除完了後、「Q」を入力すると初期画面に戻ります。

#### バージョン情報の表示

次のコマンドを実行して、Hitachi PP Installer を起動します。

```
/etc/hitachi_setup
```

Hitachi PP Installer の初期画面が表示されます。初期画面については、図 2-2 を参照してください。

初期画面で「L」を入力すると、インストール済みの日立製品の一覧が表示されます。

## 2.2.2 アンインストール

ここでは、SNMP エージェントのアンインストールの手順および注意事項について説明します。

### (1) SNMP エージェントのアンインストール手順

次の手順で SNMP エージェントをアンインストールします。

1. プログラムを終了する。
2. ユーザーファイルをバックアップする。  
SNMP エージェントのアンインストールでは、定義ファイルやログファイルなども含めて、ディレクトリごと削除されますので、必要に応じてバックアップしてください。
3. Hitachi PP Installer を実行する。  
Hitachi PP Installer の指示に従ってアンインストールしてください。Hitachi PP Installer の実行手順の詳細については、「2.2.1(3)(a) Hitachi PP Installer の使用方法」を参照してください。

### (2) アンインストール時の注意事項

SNMP エージェントをアンインストールする場合の注意事項を次に示します。

アンインストール後の不要ファイルの削除

SNMP エージェントをアンインストールしても、次に示すファイルは削除されません。不要な場合は SNMP エージェントのアンインストール後、これらのファイルを削除してください。

- /etc/SnmpAgent.d/snmpd.conf
- /etc/SnmpAgent.d/snmpd.extend
- /etc/SnmpAgent.d/esafilesys.conf
- /etc/SnmpAgent.d/esafilesys.conf.err
- /etc/SnmpAgent.d/esalocale.conf
- ユーザー指定の拡張 MIB 定義ファイル
- ユーザー指定のログファイル
- /opt/CM2/ESA/ext 配下のファイル (AIX 以外)
- /usr/CM2/ESA/ext 配下のファイル (AIX)
- /tmp/esa.log
- /tmp/.AgentSockets/ 配下のファイル
- /etc/snmpd.conf (シンボリックリンク)

## 2.2.3 バージョンアップインストール

ここでは、SNMP エージェントをバージョンアップでインストールする場合の操作について説明します。

### (1) カスタマイズした定義ファイルのバックアップ

SNMP エージェントが提供しているファイルを直接カスタマイズしている場合は、カス

## 2. SNMP エージェントの環境設定

タマイズしたファイルをバックアップしてから、バージョンアップを実施してください。バックアップについては、「3.4 バックアップとリストア」を参照してください。なお、SNMP エージェントのバージョンアップ時のインストールでは、次のファイルがすでに存在する場合には上書きされません。そのため、バックアップは不要です。

表 2-2 バージョンアップインストールで上書きされないファイル一覧

種類	パス名	ファイル名	適用 OS			
			Solaris	AIX	Linux ,Linux (IPF)	HP-UX (IPF)
構成定義ファイル	/etc/SnmpAgent.d	snmpd.conf				
	/etc/opt/OV/share/conf	snmpmib				
	/etc/opt/OV/share/conf	snmpmib.bin				
	/etc/srconf/agt	naa.cnf				-
	/opt/CM2/ESA/ext	ディレクトリ配下のファイル		-		
	/usr/CM2/ESA/ext	ディレクトリ配下のファイル	-		-	-
環境変数定義ファイル	/etc/rc.config.d	Snmp で始まるファイル		-	-	-
	/opt/CM2/ESA/opt	ディレクトリ配下のファイル	-	-		
	/usr/CM2/ESA/opt	ディレクトリ配下のファイル	-		-	-
拡張 MIB 定義ファイル	/etc/SnmpAgent.d	snmpd.extend				
コマンド	/opt/OV/bin	snmptrap				
ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafsys.conf				
動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf				

(凡例) : 上書きされない。 - : 該当なし。

## (2) SNMP トラップ送信先ポート番号の設定 (07-50 以前のバージョンから上書きインストールする場合)

SNMP エージェントを 07-50 以前のバージョンから上書きインストールする場合で SNMP 受信ポートに 161/udp 以外を指定しているときは、SNMP エージェントのインストール後に、SNMP トラップ送信先ポート番号 (SR\_TRAP\_TEST\_PORT 環境変数) に 162 を指定する必要があります。

SNMP トラップ送信先ポート番号は次の手順で設定してください。

1. snmpdm プロセスで使用される SnmpMaster ファイルに次の 2 行を追加する。

```
SR_TRAP_TEST_PORT=162
export SR_TRAP_TEST_PORT
```

2. snmpstart コマンドで SNMP エージェントを起動する。

## (3) ログ出力オプションの設定 (08-00 以前のバージョンから上書きインストールする場合)

SNMP エージェントを 08-00 以前のバージョンから上書きインストールする場合、必要なログを取得するために、上書きインストール後に次のどちらかの方法で環境変数定義ファイルを編集してください。

- 環境変数定義ファイルを直接編集する方法
- インストール用バックアップファイルの環境変数定義ファイルをコピーして編集する方法

### (a) 環境変数定義ファイルを直接編集する方法

環境変数定義ファイルを次の手順で編集してください。なお、コメント行として記載されている場合、行頭の # を削除して設定を有効にしてください。

HP-UX(IPF) の場合

1. snmpdm プロセスで使用される SnmpMaster ファイルの SNMP\_MASTER\_OPTIONS を次のように編集する。

```
SNMP_MASTER_OPTIONS="-tcplocal -aperror -apwarn -apverbose
-hexdump -vbdump"
export SNMP_MASTER_OPTIONS
```

2. extsubagt プロセスで使用される SnmpExtAgt ファイルの SNMP\_EXTAGT\_OPTIONS を次のように編集する。

```
SNMP_EXTAGT_OPTIONS="-aperror -apwarn -apverbose"
export SNMP_EXTAGT_OPTIONS
```

## 2. SNMP エージェントの環境設定

3. htc\_unixagt1 プロセスで使用される SnmpHtcunix1 ファイルの SNMP\_HTCUNIX1\_OPTIONS を次のように編集する。

```
SNMP_HTCUNIX1_OPTIONS="-aperror -apwarn -apverbose"  
export SNMP_HTCUNIX1_OPTIONS
```

4. htc\_unixagt2 プロセスで使用される SnmpHtcunix2 ファイルの SNMP\_HTCUNIX2\_OPTIONS を次のように編集する。

```
SNMP_HTCUNIX2_OPTIONS="-aperror -apwarn -apverbose"  
export SNMP_HTCUNIX2_OPTIONS
```

5. trapdestagt プロセスで使用される SnmpTrpDst ファイルの SNMP\_TRAPDEST\_OPTIONS を次のように編集する。

```
SNMP_TRAPDEST_OPTIONS="-aperror -apwarn -apverbose"  
export SNMP_TRAPDEST_OPTIONS
```

6. snmpstart コマンドで SNMP エージェントを起動する。

HP-UX(IPF) 以外の場合

1. snmpdm プロセスで使用される SnmpMaster ファイルの SNMP\_MASTER\_OPTIONS を次のように編集する。

```
SNMP_MASTER_OPTIONS="-tcplocal -aperror -apwarn -apverbose  
-hexdump -vbdump"  
export SNMP_MASTER_OPTIONS
```

2. extsubagt プロセスで使用される SnmpExtAgt ファイルの SNMP\_EXTAGT\_OPTIONS を次のように編集する。

```
SNMP_EXTAGT_OPTIONS="-aperror -apwarn -apverbose"  
export SNMP_EXTAGT_OPTIONS
```

3. htc\_unixagt1 プロセスで使用される SnmpHtcunix1 ファイルの SNMP\_HTCUNIX1\_OPTIONS を次のように編集する。

```
SNMP_HTCUNIX1_OPTIONS="-aperror -apwarn -apverbose"  
export SNMP_HTCUNIX1_OPTIONS
```

4. htc\_unixagt3 プロセスで使用される SnmpHtcunix3 ファイルの SNMP\_HTCUNIX3\_OPTIONS を次のように編集する。

```
SNMP_HTCUNIX3_OPTIONS="-aperror -apwarn -apverbose"  
export SNMP_HTCUNIX3_OPTIONS
```

5. hp\_unixagt プロセスで使用される SnmpHpunix ファイルの

SNMP\_HPUNIX\_OPTIONS を次のように編集する。

```
SNMP_HPUNIX_OPTIONS="-aperror -apwarn -apverbose"
export SNMP_HPUNIX_OPTIONS
```

6. naaagt プロセスで使用される SnmpNaa ファイルの SNMP\_NAA\_OPTIONS を次のように編集する。

```
SNMP_NAA_OPTIONS="-aperror -apwarn -apverbose -hexdump -vbdump"
export SNMP_NAA_OPTIONS
```

7. trapdestagt プロセスで使用される SnmpTrpDst ファイルの SNMP\_TRAPDEST\_OPTIONS を次のように編集する。

```
SNMP_TRAPDEST_OPTIONS="-aperror -apwarn -apverbose"
export SNMP_TRAPDEST_OPTIONS
```

8. snmpstart コマンドで SNMP エージェントを起動する。

- (b) インストール用バックアップファイルの環境変数定義ファイルをコピーして編集する方法

環境変数定義ファイルを次の手順で編集してください。

1. 旧バージョンの環境変数定義ファイルを任意の場所へ退避する。  
環境変数定義ファイルの格納ディレクトリ以外の別の場所へ退避してください。  
設定ファイルのバックアップについては「3.4 バックアップとリストア」を参照してください。
2. インストール用バックアップファイルの環境変数定義ファイルを環境変数定義ファイルのパスへコピーする。
3. 手順 1. で退避した環境変数定義ファイルと手順 2. でコピーした環境変数定義ファイルを比較して差異がある場合、退避した環境変数定義ファイルの設定内容をコピーした環境変数定義ファイルに設定する。
4. snmpstart コマンドで SNMP エージェントを起動する。

なお、環境変数定義ファイルへのパス、インストール用バックアップファイルの環境変数定義ファイルへのパスについては、「付録 A SNMP エージェントのファイルの一覧」を参照してください。

## 2.3 インストール時の注意事項

---

SNMP エージェントをインストールする場合の OS 共通の注意事項について説明します。OS 固有の注意事項については、次項以降を参照してください。

SNMP エージェントをインストールする前に、前提プログラムをインストールしてください。

SNMP エージェントを NNM と同一システムにインストールする場合は、NNM を先にインストールしてから、SNMP エージェントをインストールしてください。なお、SNMP エージェントは、SubManager と同一のシステムでは動作できません。

SNMP エージェントは、IBM RISC システム /6000 SP2 (RS/6000 SP2) にインストールできます。

IBM RISC システム /6000 SP2 (RS/6000 SP2) へのインストール/アンインストール方法については、「付録 E IBM RISC システム /6000 SP2 (RS/6000 SP2) へのインストールとアンインストール」を参照してください。

OS のバージョンアップ時の注意事項

OS を上書きでバージョンアップする場合、および AIX での Migration インストールでバージョンアップする場合、SNMP エージェントはいったんアンインストールしてください。OS のバージョンアップが終了したあとで、SNMP エージェントを再度インストールしてください。この場合には、SNMP エージェントをカスタマイズしているときは再設定が必要になります。

### 2.3.1 インストール時の注意事項 (Solaris の場合)

システムの OS が Solaris である SNMP エージェントをインストールする場合の、Solaris 固有の注意事項について説明します。OS 共通の注意事項については、「2.3 インストール時の注意事項」を参照してください。

NNM と同一ホストに SNMP エージェントをインストールする場合、次に示すディレクトリが存在することが必要です。特に、クラスタシステムの場合は注意してください。

- /var/opt/OV/share
- /etc/opt/OV/share

SNMP エージェントはインストール時に、このディレクトリ配下にファイルを作成します。ディレクトリが存在しない場合、インストールが失敗します。

Solaris のネイティブエージェントがインストールされていることを確認してください。インストールされていない場合は、OS のインストール CD-ROM からインストールしてください。

Solaris のネイティブエージェントをインストールすることで、SNMP エージェントは MIB-II の interfaces, at, ip, icmp, tcp, udp のグループ情報を Solaris のネイ

ティブエージェントから取得します。Solaris のネイティブエージェントがインストールされていない場合、SNMP エージェントが MIB-II の interfaces , at , ip , icmp , tcp , udp のグループ情報について応答しなかったり、不正な情報を応答したりします。

通常、Solaris のネイティブエージェントは Solaris をインストールするときに同時にインストールされますが、すでに運用中のシステムでネイティブエージェントがインストール済みかどうか不明な場合などには、必要に応じて確認してください。ネイティブエージェントは、次に示すパッケージから構成されます。

Solaris 9 の場合

SUNWsasnm , SUNWmibii

Solaris 10 の場合

SUNWsmagt , SUNWsasnm , SUNWmibii

なお、Solaris 10 の最新アップデート版では以下のパッケージに変更されますので、ご注意ください。

SUNWsmmgr , SUNWsmagt , SUNWsmcmd

これらのパッケージがインストールされているかどうかは次に示すコマンドで確認してください。

Solaris 9 の場合

```
/usr/bin/pkginfo SUNWsasnm SUNWmibii
```

Solaris 10 の場合

```
/usr/bin/pkginfo SUNWsmagt SUNWsasnm SUNWmibii
```

なお、Solaris 10 の最新アップデート版では次のコマンドで確認してください。

```
/usr/bin/pkginfo SUNWsmmgr SUNWsmagt SUNWsmcmd
```

SNMP エージェントのインストールで、次の OS のファイルが変更されます。

Solaris 9 の場合

/etc/init.d/init.snmpdx および /etc/snmp/conf/snmpdx.reg

Solaris 10 の場合

/etc/init.d/init.sma (SMF 機能を適用していないシステムの場合)

/lib/svc/method/svc-sma (SMF 機能を適用しているシステムの場合)

また、/etc/services ファイルに snmp 161/udp が定義されていない場合は、/etc/services ファイルに追加されます。

SNMP エージェントのインストールで、次の Solaris ネイティブエージェントの起動停止スクリプトが書き換えられます。

Solaris 9 の場合

SNMP エージェントをインストールすると、ネイティブエージェントの snmpdx プロセスの起動スクリプトファイル (/etc/init.d/init.snmpdx) が書き換えられます。/etc/init.d/init.snmpdx ファイルをカスタマイズしていた場合や SNMP エージェントインストール後に snmpdx プロセスが正しく起動しなくなった場合な

## 2. SNMP エージェントの環境設定

ど、必要に応じて `/etc/init.d/init.snmpdx` ファイルを編集してください。このとき、`snmpdx` プロセスの `-p 8161` オプションは、ネイティブエージェントアダプターが正しく動作するために必要なオプションなので、削除しないようにしてください。また、SNMP エージェントを削除すると `/etc/init.d/init.snmpdx` ファイルを SNMP エージェントインストール前の内容に復元します。この場合も必要に応じて `/etc/init.d/init.snmpdx` ファイルを編集してください。

### Solaris 10 の場合

OS のファイル (`/etc/init.d/init.sma`) を Solaris 9 の場合と同様に書き換えおよび復元をします。なお、SMF 機能を適用しているシステムの場合、OS のファイル (`/etc/init.d/svc-sma`) を Solaris 9 の場合と同様に書き換えおよび復元をします。

SMF 機能を適用していないシステムに SNMP エージェントをインストールしたあとで、SMF 機能を適用する場合、次の手順を実行してください。

1. `snmpstop` コマンドで SNMP エージェントを停止する。
2. `snmpcheck` コマンドですべてのプロセスが "not running" であることを確認する。
3. SMF 機能を適用する。  
`snmpstart` コマンド、`snmpstop` コマンド、および `snmpcheck` コマンドは変更不要です。
4. `snmpcheck` コマンドでプロセスの状態を確認し、`snmpd` プロセスおよび `snmpdx` プロセスが "running" となっている場合、次のコマンドで停止する。

```
svcadm -v disable -s svc:/application/management/snmpdx:default
svcadm -v disable -s svc:/application/management/sma:default
```

5. `snmpcheck` コマンドですべてのプロセスが "not running" であることを確認する。
6. `/lib/svc/method/svc-sma` ファイルに次の行があるか確認する。

```
/usr/sfw/sbin/snmpd udp:8161
```

上記の行がない場合、次のように編集します。

< 変更前 >

```
else
  /usr/sfw/sbin/snmpd
fi
```

< 変更後 >

```
else
  /usr/sfw/sbin/snmpd udp:8161
fi
```

7. `snmpstart` コマンドで SNMP エージェントを起動する。

8. `snmpcheck` コマンドで `extsubagt` プロセス以外のすべてのプロセスが "running" であることを確認する。  
 なお、拡張 MIB の定義を使用している場合、`extsubagt` プロセスも "running" となります。

上記の手順を実施した環境で、SNMP エージェントをアンインストールする場合、SNMP エージェントをアンインストールしたあとに、手順 6 で編集した内容を変更前の状態に戻す必要があります。

### 2.3.2 インストール時の注意事項 (AIX の場合)

システムの OS が AIX である SNMP エージェントをインストールする場合の、AIX 固有の注意事項について説明します。OS 共通の注意事項については、「2.3 インストール時の注意事項」を参照してください。

SNMP エージェントのインストールで、次のファイルが変更されます。

- `/etc/rc.tepip`
- `/etc/inittab`
- `/etc/rc.shutdown`

また、`/etc/services` ファイルに `snmp 161/udp` が定義されていない場合は、`/etc/services` ファイルに追加されます。

### 2.3.3 インストール時の注意事項 (Linux および Linux (IPF) の場合)

システムの OS が Linux または Linux (IPF) である SNMP エージェントをインストールする場合の、Linux および Linux (IPF) 固有の注意事項について説明します。OS 共通の注意事項については、「2.3 インストール時の注意事項」を参照してください。

SNMP エージェントのインストールで OS のファイルの一部が変更されます。

`/etc/services` ファイルに `snmp 161/udp` が定義されていない場合は、`/etc/services` ファイルに追加されます。

Linux または Linux (IPF) のネイティブエージェントがインストールされていることを確認してください。

Linux または Linux (IPF) のネイティブエージェントがインストールされていない場合、次の手順でネイティブエージェントをインストールしてください。

1. 次のコマンドをスーパーユーザーで実行して、ネイティブエージェントが組み込まれているかどうかを判定する。

```
#rpm -qa | grep net-snmp
```

実行結果として次の行が表示された場合、ネイティブエージェントが組み込まれています。表示されない場合、ネイティブエージェントが組み込まれていません。

`rpm` コマンドを使用してネイティブエージェントを組み込んでください。

## 2. SNMP エージェントの環境設定

```
net-snmp-x.x.x.x
net-snmp-libs-x.x.x-x
net-snmp-utils-x.x.x-x
```

### 2. OS 自動起動ファイルを確認する。

/etc/rc.d/rc3.d , /etc/rc.d/rc4.d , /etc/rc.d/rc5.d 配下の OS 自動起動ファイルに , /etc/rc.d/init.d/snmpd ファイルからシンボリックリンクが張られていることを確認します。この場合、通常、S50snmpd となっています。

シンボリックリンクが張られていない場合は、次のようにシンボリックリンクを作成してください。

```
ln -s /etc/rc.d/init.d/snmpd /etc/rc.d/rc3.d/S50snmpd
ln -s /etc/rc.d/init.d/snmpd /etc/rc.d/rc4.d/S50snmpd
ln -s /etc/rc.d/init.d/snmpd /etc/rc.d/rc5.d/S50snmpd
```

### 3. ネイティブエージェントの構成を変更する。

ネイティブエージェントはデフォルトの設定では System グループの MIB しか応答しません。すべての MIB グループを応答するように、/etc/snmp/snmpd.conf ファイルを次のように変更してください。

変更前

```
view systemview included .1.3.6.1.2.1.1
view systemview included .1.3.6.1.2.1.25.1.1
```

変更後

```
view systemview included .1.3
```

### 4. SNMP エージェントを起動する。

SNMP エージェントを起動します。マシンを再起動するか、またはスーパーユーザーで次に示す順序でコマンドを実行してください。

```
/opt/CM2/ESA/bin/snmpstop
/etc/rc.d/init.d/snmpd restart
/opt/CM2/ESA/bin/snmpstart
```

注 SNMP エージェントが稼働中の場合、SNMP エージェントを停止させてください。

Linux , Linux (IPF) での /etc/hosts の自ノードを定義してください。

SNMP エージェントはマネージャーに SNMP トラップメッセージで非同期に発生したイベントを通知します。そのメッセージ中に自ホストの IP アドレスを挿入していません。この自ホストの IP アドレスは自ホスト名に対応する IP アドレスです。

Linux または Linux (IPF) をインストールすると、/etc/hosts ファイルに自ノードの IP アドレスに対して「xxx.0.0.1」が定義されることがあります。

自ノードが「linux01」である /etc/hosts ファイルの定義例を次に示します。

```
127.0.0.1    linux01 localhost.localdomain localhost
```

上記の例の `/etc/hosts` の定義では「linux01」に対する IP アドレスは「127.0.0.1」になります。マネージャー側が UNIX 版 NNM の場合は、アラームブラウザ上の SNMP エージェントからのイベントのソース名が「127.0.0.1」となり、どのホストからメッセージが着信したのか認識できません。そこで、`/etc/hosts` ファイルには自ノードに対して「127.0.0.1」ではなく、自 IP アドレスを定義してください。`/etc/hosts` の定義例を次に示します。

```
127.0.0.1    localhost.localdomain localhost
172.16.49.18 linux01
```

マネージャー側が Windows 版 NNM の場合、NNM のデフォルトの設定では SNMP トラップメッセージ中の IP アドレスをアラームブラウザのソース名とはしません。送信された PDU の送信元 IP アドレスがアラームブラウザのソース名として表示されます。

### 2.3.4 インストール時の注意事項 (HP-UX (IPF) の場合)

システムの OS が HP-UX (IPF) である SNMP エージェントをインストールする場合の、HP-UX (IPF) 固有の注意事項について説明します。OS 共通の注意事項については、「2.3 インストール時の注意事項」を参照してください。

SNMP エージェントのインストールで OS のファイルの一部が変更されます。SNMP エージェントは 161/udp ポートを SNMP パケットの受信ポートとしています。この 161/udp ポートに接続できるプロセスは一つだけです。そこで、ネイティブエージェントと SNMP エージェントを同時に起動した場合、SNMP エージェントが 161/udp ポートに必ず接続できるように、ネイティブエージェントを OS 起動時に自動起動しない設定にしています。なお、SNMP エージェントのアンインストール時に元の名称に戻します。次に変更前と変更後のファイル名を示します。

変更前	変更後
<code>/sbin/rc1.d/K440SnmpMaster</code>	<code>/sbin/rc1.d/xK440SnmpMaster</code>
<code>/sbin/rc1.d/K435SnmpHpunix</code>	<code>/sbin/rc1.d/xK435SnmpHpunix</code>
<code>/sbin/rc1.d/K435SnmpMib2</code>	<code>/sbin/rc1.d/xK435SnmpMib2</code>
<code>/sbin/rc1.d/K435SnmpTrpDst</code>	<code>/sbin/rc1.d/xK435SnmpTrpDst</code>
<code>/sbin/rc2.d/S560SnmpMaster</code>	<code>/sbin/rc2.d/xS560SnmpMaster</code>

## 2. SNMP エージェントの環境設定

変更前	変更後
/sbin/rc2.d/S565SnmpHpunix	/sbin/rc2.d/xS565SnmpHpunix
/sbin/rc2.d/S565SnmpMib2	/sbin/rc2.d/xS565SnmpMib2
/sbin/rc2.d/S565SnmpTrpDst	/sbin/rc2.d/xS565SnmpTrpDst

また、`/etc/services` ファイルに `snmp 161/udp` が定義されていない場合は、`/etc/services` ファイルに追加されます。

NNM と同一ホストに SNMP エージェントをインストールする場合、必要なディレクトリの存在を確認してください。

NNM と同一ホストに SNMP エージェントをインストールする場合、SNMP エージェントインストール時に、次に示すディレクトリが存在することが必要です。特に、クラスタシステムの場合、注意してください。

- `/var/opt/OV/share`
- `/etc/opt/OV/share`

SNMP エージェントはインストール時にこのディレクトリ配下にファイルを作成します。ディレクトリが存在しない場合、インストールは失敗します。

## 2.4 構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) のカスタマイズ

SNMP エージェントの環境は構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) で定義します。構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) がない場合、SNMP エージェントは動作しません。

構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) に指定する内容を次に示します。

- 連絡先と設置場所
- コミュニティ名
- トラップのあて先

構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) は SNMP エージェントで提供されています。提供されている構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) には定義についての情報が記述されています。必要に応じてカスタマイズしてご使用ください。

構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) の提供内容については、「2.4.4 構成定義ファイルの形式」を参照してください。

### 2.4.1 連絡先と設置場所の設定

連絡先はシステムを管理する上での連絡先、およびシステム管理者に連絡する方法についての情報です。設置場所はシステムの物理的な位置を示します。

連絡先と設置場所は、次のどちらかの方法で設定できます。

- 構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf )
- snmpdm プロセスのオプション

二つの方法が併用されている場合、システムは snmpdm プロセスのオプションで指定された値を使用します。

#### (1) 構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) で設定する方法

次に示す手順で構成定義ファイル ( /etc/SnmpAgent.d/snmpd.conf ) を編集して、連絡先と設置場所を設定します。

1. 構成定義ファイルの次の 2 行を検索する。

```
#contact:      # enter contact person for agent
#location:     # enter location of agent
```

なお、検索する行は、構成定義ファイルの最後に記載されています。

## 2. SNMP エージェントの環境設定

2. contact: ラベルの前の # と、コメント (# のあとの部分) を削除する。  
location: ラベルも同様にします。
3. contact: ラベルのあとに連絡先を ASCII 文字列で記述する。  
連絡先の最大長は 255 文字です。なお、連絡先には連絡方法の情報も付加します。
4. location: ラベルのあとに設置場所を ASCII 文字列で記述する。  
設置場所の最大長は 255 文字です。

(例)

```
contact: Bob Jones (Phone 555-2000)
location: 1st Floor near Mens Room
```

### (2) snmpd プロセスのオプションで設定する方法

オプションに連絡先と設置場所を指定して、snmpd プロセスを起動します。snmpd プロセスの詳細については、「5. コマンドとプロセス」の「snmpd」を参照してください。

(例)

AIX の場合の指定例を次に示します。

```
/usr/sbin/snmpd -C 連絡先 -L 設置場所
```

## 2.4.2 コミュニティ名の設定

コミュニティ名とは、SNMP プロトコルで MIB 値にアクセスするために必要なパスワードです。コミュニティ名のセキュリティは低く、ネットワーク上でオープンに使用されています。

次のような場合に、SNMP エージェントのコミュニティ名を利用すると便利です。

- get コミュニティ名を指定して、不正なマネージャーからの SNMP 要求に対する MIB 値の参照を抑制したい。
- set コミュニティ名を指定して、不正なマネージャーからの SNMP 要求に対する MIB 値の更新を抑制したい。

マネージャーと SNMP エージェントのコミュニティ名は、構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) で設定します。

### (1) コミュニティ名の種類

コミュニティ名には、get コミュニティ名および set コミュニティ名があります。これらのコミュニティ名を使い分けることによって、どの SNMP 要求に回答するかを決定できます。

### get コミュニティ名

GetRequest に対するパスワードです。

get コミュニティ名を使用すると、GetRequest だけに応答します。

### set コミュニティ名

GetRequest および SetRequest の両方に対するパスワードです。

set コミュニティ名を使用すると、GetRequest および SetRequest の両方に応答します。

コミュニティ名は複数登録できます。

## (2) コミュニティ名の登録方法

SNMP エージェントのインストール時には、get コミュニティ名および set コミュニティ名に「public」が設定されています。

デフォルトの get コミュニティ名を変更する場合の手順を次に示します。

1. get コミュニティ名の登録
2. set コミュニティ名の登録
3. マネージャーへのコミュニティ名の格納
4. ネイティブエージェントのコミュニティ名設定 (Solaris の場合)

### (a) get コミュニティ名の登録

get コミュニティ名を登録する手順を次に示します。

1. 構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) の次の行を検索する。

```
get-community-name:    public
```

なお、get-community-name: ラベルはファイルの最終行の近くにあります。

2. get コミュニティ名を変更する。

提供時は「public」が設定されています。変更する場合、「public」を削除して、SNMP エージェントの get コミュニティ名を ASCII 文字列で記述します。複数の get コミュニティ名を指定したい場合は、行を追加して定義できます。コミュニティ名の指定方法の詳細については、「2.4.2(3) コミュニティ名の指定方法」を参照してください。

(例)

```
get-community-name: public
get-community-name: private
```

## 2. SNMP エージェントの環境設定

### 参考

---

すべての SNMP エージェントに同一の get コミュニティ名を使用すると便利です。

---

#### (b) set コミュニティ名の登録

set コミュニティ名を登録する手順を次に示します。

1. 構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) の次の行を検索する。

```
#set-community-name:      # enter community name
```

なお、#set-community-name: ラベルはファイルの最終行の近くにあります。

2. set コミュニティ名を追加する。

提供時はコメント行となっています。set-community-name: ラベルの前の # と、コメント (# のあとの部分) を削除して、set-community-name: ラベルのあとに、SNMP エージェントの set コミュニティ名を ASCII 文字列で記述します。複数の set コミュニティ名を指定したい場合は、行を追加して定義します。

コミュニティ名の指定方法の詳細については、「2.4.2(3) コミュニティ名の指定方法」を参照してください。

(例)

```
set-community-name: private
set-community-name: point
```

#### 注意事項

- コミュニティ名の指定について  
get コミュニティ名と set コミュニティ名を同じ名称にする場合には、set-community-name: ラベルだけに指定してください。get コミュニティ名と set コミュニティ名を別の名称にする場合には、get-community-name: ラベルと set-community-name: ラベルにそれぞれコミュニティ名を指定してください。
- 認証失敗トラップの送信抑止について  
SNMP エージェントのコミュニティ名を設定したあとで、認証失敗トラップの送信を抑止したい場合、構成定義ファイル (/etc/srconf/agt/snmpd.cnf) の snmpEnableAuthenTraps に「2」を設定してから SNMP エージェントを再起動してください。  
認証失敗トラップの送信については、「(4) 認証失敗トラップの送信」を参照してください。

#### (c) マネージャーへのコミュニティ名の格納

マネージャーのアプリケーションが各 SNMP エージェントの正しいコミュニティ名を使用して MIB にアクセスできるようにします。

## (d) ネイティブエージェントのコミュニティ名設定 (Solaris の場合)

SNMP エージェントが稼働するシステムが Solaris の場合、構成定義ファイル (/etc/snmp/conf/snmpd.conf) にネイティブエージェントのコミュニティ名の設定が必要です。

ネイティブエージェントのコミュニティ名を設定する手順を次に示します。

## 1. ネイティブエージェントのコミュニティ名が設定されていることを確認する。

Solaris 9 の場合

構成定義ファイル (/etc/snmp/conf/snmpd.conf) に、次の 2 行が定義されていることを確認してください。

```
system-group-read-community public
read-community public
```

Solaris 10 の場合

構成定義ファイル (/etc/sma/snmp/snmpd.conf) の次の行が定義されていることを確認してください。

```
rocommunity public
```

## 2. 定義されていない場合、ネイティブエージェントのコミュニティ名を追加する。

構成定義ファイル (/etc/sma/snmp/snmpd.conf) に手順 1 の行を追加してください。

## (3) コミュニティ名の指定方法

コミュニティ名の指定方法について説明します。

## (a) get コミュニティ名

get コミュニティ名は、構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) の次に示す行に指定します。

```
get-community-name: getコミュニティ名 オプション
```

注 オプションの詳細については、「(c) オプション」を参照してください。

- get コミュニティ名を指定すると、指定したコミュニティ名以外からの要求は認証失敗になります。
- get コミュニティ名の指定がない場合、GetRequest に応答しません。ただし、set コミュニティ名が指定されていれば、set コミュニティ名を使用した GetRequest には応答します。
- get コミュニティ名を指定する際は、「:(コロン)」の後ろに、必ず半角スペースを入れてください。

## 2. SNMP エージェントの環境設定

- 構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) に複数の get コミュニティ名を設定することで、SNMP エージェントは複数の get コミュニティ名に応答できます。

### (b) set コミュニティ名

set コミュニティ名は、構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) の次に示す行で指定します。

```
set-community-name: setコミュニティ名 オプション
```

注 オプションの詳細については、「(c) オプション」を参照してください。

- set コミュニティ名の指定がない場合、SNMP エージェントは SetRequest に応答しません。
- set コミュニティ名を指定する際は、「:( コロン)」の後ろに、必ず半角スペースを入れてください。
- マネージャーが MIB 値を設定するためには、set コミュニティ名を設定する必要があります。マネージャーは登録済みの set コミュニティ名を使用して、MIB 値を設定します。
- 複数の set コミュニティ名に応答するように、SNMP エージェントを構成できます。

### 注意事項

get-community-name: ラベルと set-community-name: ラベルに同じ名称を指定すると、構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) の下の方に記述された行が有効になります。上から set-community-name: ラベル、get-community-name: ラベルの順に記述すると、その名称は get コミュニティ名として指定されたものと解釈され、GetRequest にだけ使用できるコミュニティ名となります。この場合、SetRequest でその名前を使用すると認証失敗となります。GetRequest と SetRequest に同じ名称を使用したい場合は、set-community-name: ラベルだけを指定してください。

### (c) オプション

オプションで IP: と VIEW: が指定できます。

両方を省略するとコミュニティ名はすべての IP アドレスからの要求を許可します。また、SNMP エージェントがサポートしているすべての MIB にアクセスできます。

#### IP:

SNMP 要求中のコミュニティ名で MIB にアクセス可能な IP アドレスを制限します。MIB にアクセスできる IP アドレスを空白で区切って記述します。ホスト名は指定できません。コミュニティ名と IP:、および IP: と IP アドレスの間には必ず 1 文字以上の空白を挿入してください。

(例)

```
get-community-name: public IP: 172.16.45.17 172.16.45.18
```

SNMP エージェントは SNMP 要求中のコミュニティ名が public の場合、172.16.45.17、172.16.45.18 からの SNMP 要求だけに応答します。

#### VIEW:

指定されたコミュニティ名でアクセス可能な MIB を制限します。アクセス可能なサブツリーのオブジェクト ID (mib-2 の場合の例 1.3.6.1.2.1) を空白で区切って記述します。オブジェクト ID の先頭に「-」を付加するとそのサブツリーにはアクセスできなくなります。コミュニティ名と VIEW:、および VIEW: とオブジェクト ID の間には 1 文字以上の空白を挿入してください。また、「-」の前に 1 文字の空白を挿入してください。

(例)

```
get-community-name: public VIEW: 1.3.6.1.2.1 -1.3.6.1.2.1.1
```

SNMP エージェントは SNMP 要求中のコミュニティ名が public の場合、1.3.6.1.2.1.1 を除いた 1.3.6.1.2.1 配下の MIB のアクセスを許可します。

#### 注意事項

- IP: と VIEW: の同時指定

IP: と VIEW: を同時に指定する場合は IP: を先に指定してください。また、IP: と VIEW: は 1 行に記述し、改行はしないでください。set-community-name: にも同様に IP: と VIEW: が指定できます。

- AIX の場合の VIEW: 指定

ネイティブエージェントが起動時にオブジェクト ID 「1.3.6.1.4.1.2」配下を参照します。そのため、AIX 5L V5.2 以降で VIEW オプションを設定する場合は、「1.3.6.1.4.1.2」配下の MIB のアクセスを許可するように設定してください。

(例)

mib-2 配下をアクセス可能な MIB とする場合

```
get-community-name:public VIEW: 1.3.6.1.2.1 1.3.6.1.4.1.2
```

- AIX の場合の IP: 指定

mib-2 配下の MIB を取得する際に、OS 提供プロセスから SNMP エージェントに対して SNMP リクエストを送信します。そのため、IP オプションで MIB を応答する IP アドレスを制限した場合、自ホストの IP アドレスを定義する必要があります。AIX の OS 提供プロセスと、SNMP エージェントの関係の詳細については、「3.5.2 運用上の注意事項 (AIX の場合)」を参照してください。

(例)

マネージャーマシンの IP アドレス : 172.16.45.17

SNMP エージェントマシンの IP アドレス : 172.16.50.15

上記の IP アドレスを使用して、マネージャーマシンからの SNMP リクエストにだけ応答を制限する場合

## 2. SNMP エージェントの環境設定

```
get-community-name: public IP: 172.16.45.17 172.16.50.15
```

### (4) 認証失敗トラップの送信

マネージャーが誤りまたは無効のコミュニティ名を SNMP エージェントに送信した場合は、認証失敗になります。誤りまたは無効のコミュニティ名を受信すると、認証失敗トラップをマネージャーに送信します。

GetRequest に対して get コミュニティ名の指定がない場合、SNMP エージェントはどんなコミュニティ名にも応答しません。その場合、SNMP エージェントは認証失敗のトラップを送信します。

## 2.4.3 トラップのあて先の設定

SNMP エージェントはトラップのあて先を持っています。トラップのあて先とは、SNMP トラップをどこに送信すればよいかを決め、SNMP エージェントのトラップを受け取るマネージャーを確認するものです。SNMP エージェントを管理するマネージャーが複数の場合は、SNMP エージェントが持つトラップのあて先も複数です。

SNMP エージェントを管理するマネージャー側製品が NNM および SubManager の場合と NNMi および任意のマネージャーの場合について、トラップのあて先の設定について説明します。

### (1) NNM および SubManager の場合

NNM および SubManager では、netmon プロセスが SNMP を使用して自動的に SNMP エージェントのトラップあて先を追加します。しかし、SNMP エージェントのマネージャー側の製品として JP1/Cm2/NNM バージョン 05-20 以降、または HP NNM バージョン 6 以降を使用する場合は、SNMP エージェントと NNM のデフォルトの set コミュニティ名が一致していないため、トラップのあて先は設定されません。netmon プロセスから SNMP を使用して自動的に SNMP エージェントのトラップあて先を追加したい場合は、SNMP エージェントと JP1/Cm2/NNM または HP NNM との set コミュニティ名を一致させてください。コミュニティ名の登録方法については、「2.4.2 コミュニティ名の設定」を参照してください。

ノードを管理対象から外した場合、またはノードをマップから削除した場合は、NNM および SubManager の netmon プロセスが自動的に構成定義ファイル (`/etc/SnmpAgent.d/snmpd.conf`) の SNMP エージェントのトラップのあて先を削除します。

### (2) NNMi および任意のマネージャーの場合

SNMP エージェントが NNMi および任意のマネージャーにトラップを送信したい場合は、ユーザーがトラップのあて先を設定する必要があります。SNMP エージェントのトラップのあて先は構成定義ファイル (`/etc/SnmpAgent.d/snmpd.conf`) を編集して設定します。次に手順を示します。

1. 構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) で次の行を検索する。

```
#trap-dest: # enter trap destination
```

なお、#trap-dest: ラベルはファイルの最終行の近くにあります。

2. trap-dest: ラベルの前の # と、コメント (#以降の部分) を削除する。
3. trap-dest: ラベルのあとに、SNMP エージェントがトラップを送信するマネージャーのホスト名または IP アドレスを記述する。

(例)

```
trap-dest: 15.2.113.223
```

4. トラップのあて先をさらに追加する場合は、trap-dest: の行を追加する。

## 2.4.4 構成定義ファイルの形式

構成定義ファイル (/etc/SnmpAgent.d/snmpd.conf) の提供内容を次に示します。

```
# Use this file (snmpd.conf) to configure the following SNMP Agent
# parameters. The valid configuration keywords are:
#
#   get-community-name:
#   set-community-name:
#   trap-dest:
#   contact:
#   location:
#
# get-community-name: <name> IP: <ip_address_list> VIEW:
<view_list>
#
# The agent will only respond to get requests using <name> as
# the community name. Embedded blanks are not allowed.
# If a community name is not specified in either this file or
# snmpd.cnf, the agent not respond to any get requests.
# More than one community name can be configured for the agent
# by adding a separate entry for each name to be allowed. For
# example,
#
#       get-community-name: secret
#
# restricts access to only those requests using community "secret";
# and,
#
#       get-community-name: secret
#       get-community-name: private
#
# restricts access to only those requests using either community
# "secret" or "private".
#
# The IP: and VIEW: qualifiers are optional. When either or
# both qualifiers are omitted, the community name is allowed
```

## 2. SNMP エージェントの環境設定

```
# for any requesting IP address and provides access the entire
# MIB supported by the agent, respectively.
#
# The IP: qualifier further restricts use of the community name
# to only those requests that originate from one of the listed
# IP addresses. Host names are not supported. For example,
#
#     get-community-name: operator IP: 15.2.112.90 15.2.114.101
#
# only allows access using community "operator" from IP address
# 15.2.112.90 or 15.2.114.101.
#
# The VIEW: qualifier further restricts access using the community
# name to the sub-set of the agent's supported MIB identified by
# the space list of "MIB view sub-trees". A view sub-tree may be
# either the object identifier (1.3.6.1.2.1.1) or object name
# (system) of the MIB sub-tree to be included. The '-' character
# may be used to exclude an oid/name from the view. For example,
#
#     get-community-name: operator VIEW: 1.3.6.1.2.1 1.3.6.1.4.1.11
# -1.3.6.1.2.1.1
#
# allows access using community "operator" to all MIB objects
# under "mib-2" except those objects under "system", plus all
# objects under the "hp" sub-tree.
#
# For example,
#
#     get-community-name: operator IP: 15.2.112.90 15.2.114.101
# VIEW: 1.3.6.1.2.1 1.3.6.1.4.1
#
# combines the access restrictions described in the previous
# examples for community "operator".
#
# set-community-name: <name> IP: <ip_address_list> VIEW:
# <view_list>
#
# The agent will only process get or set requests using <name>
# as the community name. Embedded blanks are not allowed.
# If a set community name is not configured in either this file
# or snmpd.cnf, set requests are not allowed by the agent. More
# than one set community name can be configured by adding a
# separate entry for each name to be allowed. For example,
#
#     set-community-name: control
#
# enables the agent to process set requests using the community
# name "control".
#
# The IP: and VIEW: qualifiers are optional. They provide the
# same access restrictions for the set community name as described
# above for get-community-name. For example,
#
#     set-community-name: technician VIEW: system
#     set-community-name: administrator IP: 15.2.112.90
#
# allows set request processing using community "technician" to
```

```
# only those objects under the "system" group (e.g., sysContact);
# and allows set requests processing on any object using community
# "administrator", but only from IP address 15.2.112.90.
#
# trap-dest: <trap destination>
#
# Specifies the system name where traps will be sent. The system
# name is usually the hostname or IP address of the management
# station. More than one trap destination can be configured by
# by adding a separate entry for each destination. For example,
#
#     trap-dest: manager1
#     trap-dest: 15.2.113.223
#
# will cause traps to be sent to the systems named "manager1,"
# and "15.2.113.223".
#
# contact: <contact string>
#
# Specifies the value of the MIB-II sysContact object. This
# information usually includes the name of person responsible
# for the agent system, plus information on how to contact this
# person. For example, if Bob Jones is responsible person, and
# his phone number is 555-2000, enter:
#
#     contact: Bob Jones (Phone 555-2000)
#
# NOTE: the value specified in this file overrides snmpd.cnf
#
# location: <location string>
#
# Specifies the value of the MIB-II sysLocation object. For
# example, if the agent system is on the first floor near the
# mens room, enter:
#
#     location: 1st Floor near Mens Room
#
# NOTE: the value specified in this file overrides snmpd.cnf
#
##
#
get-community-name:    public
#set-community-name:  # enter community name
#contact:             # enter contact person for agent
#location:            # enter location of agent
#trap-dest:           # enter trap destination
```

## 2.5 ネイティブエージェントアダプターの設定 ( HP-UX (IPF) 以外の場合 )

この節では、ネイティブエージェントアダプターの機能、ネイティブエージェントおよびネイティブエージェントアダプターの設定について説明します。

### 2.5.1 ネイティブエージェントアダプターの機能

ネイティブエージェントアダプターは、ネイティブエージェントと SNMP エージェントを接続するためのアダプター機能です。この機能を使用すると、ネイティブエージェントが提供する MIB オブジェクトと、mib-2 配下にある system および snmp 以外のグループの MIB も SNMP を通じて取得できます。ネイティブエージェントアダプターとネイティブエージェントの間は、SNMPv1 で通信します。

ネイティブエージェントから取得する MIB グループの一覧を次の表に示します。

表 2-3 ネイティブエージェントから取得する MIB グループの一覧

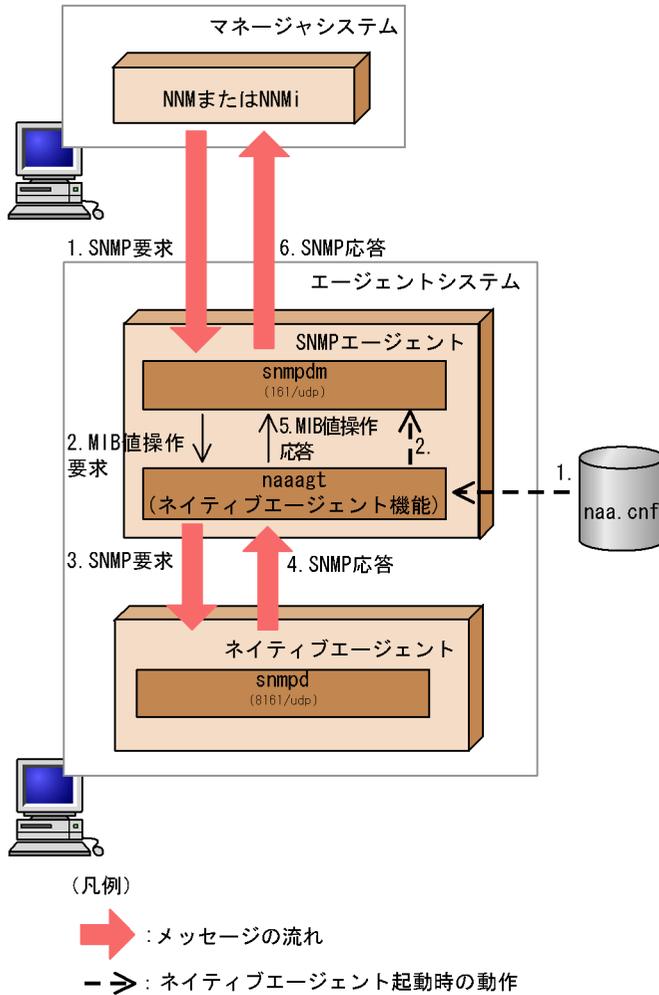
MIB グループ名称	MIB オブジェクト	read/write	適用 OS		
			Solaris	AIX	Linux , Linux (IPF)
interfaces	.1.3.6.1.2.1.2	read			
at	.1.3.6.1.2.1.3	read			
ip	.1.3.6.1.2.1.4	read			
icmp	.1.3.6.1.2.1.5	read			
tcp	.1.3.6.1.2.1.6	read			
udp	.1.3.6.1.2.1.7	read			
host	.1.3.6.1.2.1.25	read	-		

( 凡例 )

- : 取得できる。
- : 取得できない。

ネイティブエージェントアダプターの動作を次の図に示します。

図 2-3 ネイティブエージェントアダプターの動作



注 1 マスターエージェントが SNMP 要求受信用ポートとして、次の UDP ポートをバインドして起動します。

- Solaris , AIX の場合 : UDP ポート 161
- Linux , Linux (IPF) の場合 : UDP ポート 22161

注 2 ネイティブエージェントが SNMP 要求受信用ポートとして、次の UDP ポートをバインドして起動します。

- Solaris , AIX の場合 : UDP ポート 8161
- Linux , Linux (IPF) の場合 : UDP ポート 161

次に、ネイティブエージェントアダプターの起動時、および NNM または NNMi からの SNMP 要求時の動作について説明します。なお、動作の番号は図 2-3 での番号を示しま

## 2. SNMP エージェントの環境設定

す。

### ネイティブエージェントアダプター起動時の動作

1. ネイティブエージェントアダプターは起動時に `naa.cnf` 定義ファイルを読み込みます。
2. ネイティブエージェントアダプターは、`naa.cnf` 定義ファイルに指定された MIB オブジェクトを、ネイティブエージェントアダプターが処理する MIB オブジェクトとして、マスターエージェントに登録します。

### NNM または NNMi からの SNMP 要求時の動作

1. NNM または NNMi から `naa.cnf` 定義ファイルに定義した MIB オブジェクトについての SNMP 要求が送信されます。
2. マスターエージェントは、MIB 値操作要求 (リクエスト) をネイティブエージェントアダプターに配信します。
3. ネイティブエージェントアダプターは、SNMP 要求を SNMP パケットに再構築して、UDP ポート (Solaris, AIX の場合は 8161, Linux, Linux (IPF) の場合は 161) に送信します。UDP ポートに送信された要求はネイティブエージェントに受信されます。
4. ネイティブエージェントは SNMP 応答をネイティブエージェントアダプターに送信します。ネイティブエージェントにとっては、ネイティブエージェントアダプターは SNMP マネージャーのように働きます。
5. ネイティブエージェントアダプターはマスターエージェントに MIB 値操作応答を返します。
6. マスターエージェントは、NNM または NNMi に SNMP 応答を返します。

ネイティブエージェントアダプターの対象となるネイティブエージェントを次の表に示します。

表 2-4 ネイティブエージェントアダプターの対象となるネイティブエージェント

適用 OS	対象となるネイティブエージェント
Solaris 9	<code>/usr/lib/snmp/snmpdx</code>
Solaris 10	<code>/usr/sfw/sbin/snmpd</code> <code>/usr/lib/snmp/snmpdx</code>
AIX	<code>/usr/sbin/snmpd</code>
Linux, Linux (IPF)	<code>/usr/sbin/snmpd</code>

### 注

Linux, Linux (IPF) では、ネイティブエージェントは、CD-ROM で提供されているか、OS インストール時にインストールするかを選択できます。

## 2.5.2 ネイティブエージェントの設定 (AIX, Solaris の場合)

ネイティブエージェントと SNMP エージェントを同時に動作させ、ネイティブエージェントアダプターを使ってネイティブエージェントの MIB を取得できるようにするには、ネイティブエージェントがポート 8161 を SNMP 要求の受信ポートとして使用するよう設定する必要があります。通常、SNMP エージェントのインストール時に自動的に設定されます。

## 2.5.3 ネイティブエージェントアダプターの設定方法

ネイティブエージェントアダプターを設定するには、naa.cnf 定義ファイル (/etc/srconf/agt/naa.cnf) を編集します。編集後、SNMP エージェントを起動すると設定が有効になります。

ファイルの各行のフォーマットを次に示します。

タグ 値
------

タグにはキーワードを、値にはキーワードに対応する設定値を指定します。

naa.cnf 定義ファイルに指定する内容を次に示します。

- ネイティブエージェントの MIB オブジェクトの登録
  - 読み取り専用 MIB オブジェクトの追加
  - 読み書き可能 MIB オブジェクトの追加
- ネイティブエージェントのコミュニティ名
  - GET リクエストで使用するコミュニティ名の指定
  - SET リクエストで使用するコミュニティ名の指定

### (1) ネイティブエージェントの MIB オブジェクトの登録

ネイティブエージェントから取得、または設定したい MIB オブジェクトを naa.cnf 定義ファイルに設定します。取得する場合は、タグに read を指定し、値に MIB のオブジェクト ID を指定します。設定する場合は、タグに write を指定し、値に MIB のオブジェクト ID を指定します。

デフォルトで naa.cnf 定義ファイルに定義されている MIB オブジェクト ID を次の表に示します。

## 2. SNMP エージェントの環境設定

表 2-5 naa.cnf 定義ファイルにデフォルトで定義されている MIB オブジェクト ID

MIB オブジェクト ID	read/write	適用 OS		
		Solaris	AIX	Linux , Linux (IPF)
.1.3.6.1.2.1.2	read			
.1.3.6.1.2.1.3	read			
.1.3.6.1.2.1.4	read			
.1.3.6.1.2.1.5	read			
.1.3.6.1.2.1.6	read			
.1.3.6.1.2.1.7	read			
.1.3.6.1.2.1.10	read	-		-
.1.3.6.1.2.1.12	read	-		-
.1.3.6.1.2.1.25	read	-		
.1.3.6.1.4.1.42	read		-	-
.1.3.6.1.4.1.2	read	-		-
.1.3.6.1.4.1.4	read	-		-
.1.3.6.1.4.1.2021	read	-	-	

### (凡例)

- : naa.cnf 定義ファイルに定義あり。
- : naa.cnf 定義ファイルに定義なし。

### 注

Solaris 10 でネイティブエージェントから「.1.3.6.1.2.1.25」と「.1.3.6.1.4.1.2021」グループの MIB を取得できます。必要があれば naa.cnf 定義ファイルにデフォルトで定義されている MIB オブジェクトに追加してください。

### 読み取り専用 MIB オブジェクトの追加

MIB のサブツリー、または個別の MIB オブジェクトを読み取り専用として設定するには、naa.cnf 定義ファイルに、read タグの定義行を追加します。追加する行のフォーマットを次に示します。

read オブジェクトID

オブジェクト ID は、MIB のサブツリー、または個別の MIB オブジェクトのオブジェクト識別子です。オブジェクト ID は、数値形式で指定します。オブジェクトの名称は使用できません。オブジェクト ID に接尾辞を指定してもエラーにはなりません。なお、オブジェクト ID の最初の「.(ドット)」は指定しないでください。

(例)

naa.cnf 定義ファイルが次のような定義の場合、ネイティブエージェントアダプターは、MIB-II の interfaces, at, ip, icmp, tcp, udp, host の各グループをネイティブエージェントから取得します。

```
read 1.3.6.1.2.1.2
read 1.3.6.1.2.1.3
read 1.3.6.1.2.1.4
read 1.3.6.1.2.1.5
read 1.3.6.1.2.1.6
read 1.3.6.1.2.1.7
read 1.3.6.1.2.1.25
```

#### 読み書き可能 MIB オブジェクトの追加

個別の MIB オブジェクトを読み書き可能として設定するには、naa.cnf 定義ファイルに、write タグの定義行を追加します。追加する行のフォーマットを次に示します。

```
write オブジェクトID
```

オブジェクト ID は、個別の MIB オブジェクトのオブジェクト識別子です。オブジェクト ID は、数値形式で指定します。オブジェクトの名称は使用できません。また、オブジェクト ID に接尾辞を指定してもエラーにはなりません。なお、オブジェクト ID の最初の「.(ピリオド)」は指定しないでください。

MIB のサブツリーを読み書き可能として登録することもできますが、この方法はお勧めしません。

(例)

naa.cnf 定義ファイルが次のような定義の場合、ネイティブエージェントアダプターは、サブツリー enterprises.116 の下の MIB オブジェクトをネイティブエージェントに登録します。

```
write 1.3.6.1.4.1.116
```

## (2) ネイティブエージェントのコミュニティ名

ネイティブエージェントアダプターがネイティブエージェントに対して送信する GET リクエスト、または SET リクエストを naa.cnf 定義ファイルに設定します。GET リクエストの場合は、タグに readcomm を指定し、値にコミュニティ名を指定します。SET リクエストの場合は、タグに writecomm を指定し、値にコミュニティ名を指定します。

#### GET リクエストで使用するコミュニティ名の指定

ネイティブエージェントアダプターがネイティブエージェントに対して送信する GET リクエスト中のコミュニティ名 (デフォルトは「public」) を設定するには、naa.cnf 定義

## 2. SNMP エージェントの環境設定

ファイルに `readcomm` タグの定義行を追加します。追加する行のフォーマットを次に示します。

```
readcomm コミュニティ名
```

GET リクエストで使用するコミュニティ名を指定する場合の注意事項を次に示します。

- `readcomm` タグとコミュニティ名の間にはスペースを 1 文字挿入してください。
- コミュニティ名の最大長は 60 文字です。
- `naa.cnf` 定義ファイル内で `readcomm` タグは 1 個だけ指定してください。
- `readcomm` タグを使用する場合、`naaagt` プロセスの `-readcomm` オプションは指定しないでください。

(例)

`naa.cnf` 定義ファイルで次のように定義している場合、ネイティブエージェントアダプターはネイティブエージェントに対して GET リクエスト中のコミュニティ名に「`snmpread`」を使用します。

```
readcomm snmpread
```

SET リクエストで使用するコミュニティ名の指定

ネイティブエージェントアダプターがネイティブエージェントに対して送信する SET リクエスト中のコミュニティ名 (デフォルトは「`public`」) を設定するには、`naa.cnf` 定義ファイルに `writcomm` タグの定義行を追加します。追加する行のフォーマットを次に示します。

```
writcomm コミュニティ名
```

SET リクエストで使用するコミュニティ名を指定する場合の注意事項を次に示します。

- `writcomm` タグとコミュニティ名の間にはスペースを 1 文字挿入してください。
- コミュニティ名の最大長は 60 文字です。
- `naa.cnf` 定義ファイル内で `writcomm` タグは 1 個だけ指定してください。
- `writcomm` タグを使用する場合、`naaagt` プロセスの `-writcomm` オプションは指定しないでください。

(例)

`naa.cnf` 定義ファイルが次に示す定義の場合、ネイティブエージェントアダプターはネイティブエージェントに対して SET リクエスト中のコミュニティ名に「`snmpwrite`」を使用します。

```
writecomm snmpwrite
```

## 2.5.4 ネイティブエージェントアダプターの使用上の注意事項

ネイティブエージェントアダプターを使用する場合の注意事項を次に示します。

### (1) ネイティブエージェントの再起動時の注意事項

Solaris および AIX の場合、ネイティブエージェントアダプターを使用していて、個別にネイティブエージェントを再起動するとき、ネイティブエージェントが UDP ポート 8161 をバインドするように再起動しなければなりません。このような場合、ネイティブエージェントの起動、停止コマンドの代わりに、snmpstop コマンド、snmpstart コマンドを使用してください。snmpstop コマンドはネイティブエージェントを停止し、snmpstart コマンドはネイティブエージェントを起動します。

### (2) naa.cnf 定義ファイル削除時の注意事項

naa.cnf 定義ファイルを削除した状態でネイティブエージェントアダプターを起動しないでください。naa.cnf 定義ファイルを削除してネイティブエージェントアダプターを起動すると、ネイティブエージェントから MIB-II 情報だけを取得するように動作します。ただし、ネイティブエージェントから MIB-II 情報だけを取得するように設定したい場合でも、この使用方法はお勧めしません。取得する MIB オブジェクトを naa.cnf 定義ファイルに指定するようにしてください。

### (3) naaagt の SNMP パケット送信ポート変更時の注意事項

Solaris および AIX の場合、UDP ポート 8161 がすでにほかのプログラムに使用されている場合は、SNMP エージェントを停止した状態で、naaagt プロセスの SNMP パケット送信ポートとネイティブエージェントの SNMP 受信ポートを、そのシステムで使用していないポート番号に変更してください。naaagt プロセスの SNMP パケット送信ポートは naaagt プロセスの `-port` オプションを指定します。ネイティブエージェントの SNMP 受信ポートを変更する方法を次に示します。変更後は snmpstart コマンドを使用して、SNMP エージェントを再起動してください。

AIX 5L V5.2/V5.3 の場合

「3.2 SNMP エージェントの SNMP 受信ポートの変更」を参照してください。

Solaris 9 の場合

```
/etc/init.d/init.snmdx ファイルの /usr/lib/snmp/snmpdx -p 8161 -y -c /etc/snmp/conf  
の行の -p オプションの値を変更してください。
```

Solaris 10 の場合

## 2. SNMP エージェントの環境設定

/etc/init.d/init.sma ファイルの次の行を変更してください。

```
prog="/usr/sfw/sbin/snmpd udp:8161"
```

### (4) naa.cnf 定義ファイルの仕様についての注意事項 (Solaris の場合)

Solaris 版 SNMP エージェントの naaagt プロセスが参照する naa.cnf 定義ファイルと、NNM が提供する naaagt プロセスが参照する naa.cnf 定義ファイルとは、パス名および定義の仕様が異なります。NNM が提供する naa.cnf 定義ファイルを SNMP エージェントでそのまま使用することはできません。SNMP エージェントの naa.cnf 定義ファイルの仕様については、「2.5.3 ネイティブエージェントアダプターの設定方法」を参照してください。

## 2.6 拡張 MIB オブジェクトの定義

---

拡張 MIB オブジェクトは次のどちらかの拡張 MIB 定義ファイルで定義します。

- /etc/SnmpAgent.d/snmpd.extend ファイル
- /opt/Cm2/ESA/est ディレクトリ配下の拡張 MIB 定義ファイル。なお、拡張 MIB 定義ファイルの拡張子は def を指定してください。

このマニュアルでは、拡張 MIB 定義ファイル名として /etc/SnmpAgent.d/snmpd.extend を使用して、拡張 MIB オブジェクトの定義の手順を説明しています。/opt/Cm2/ESA/est ディレクトリ配下に拡張 MIB 定義ファイルを格納する場合、複数の拡張 MIB 定義ファイルを追加する方法と同じです。「2.6.10 複数の拡張 MIB 定義ファイルの設定」を参照してください。

SNMP エージェントに定義された MIB オブジェクトにマネージャーがアクセスするには、次に示す RFC で指定された規則に従って MIB モジュールを記述します。

- RFC1155:Structure and Identification of Management Information for TCP/IP-based Internets
- RFC1212:Concise MIB Definitions

拡張 MIB オブジェクトを定義する手順と参照先を次の図に示します。

図 2-4 拡張 MIB オブジェクトの定義の流れ



(凡例) ▼ : 必要な作業    ▽ : 任意の作業

なお、複数の拡張 MIB 定義ファイルを SNMP エージェントに設定できます。複数の拡張 MIB 定義ファイルを定義する場合の手順については、「2.6.10 複数の拡張 MIB 定義ファイルの設定」を参照してください。

SNMP エージェントでは、拡張 MIB 定義のサンプルとして、`/opt/OV/prg_samples/eagent/snmpd.extend` ファイルを提供しています。また、MIB オブジェクトの定義および拡張 MIB 定義ファイル (`/etc/SnmpAgent.d/snmpd.extend`) の作成例を、「2.6.11 拡

張 MIB オブジェクトの定義例」に記載しています。

## 2.6.1 MIB オブジェクトの定義

MIB モジュールには、一つ以上の MIB オブジェクトを定義でき、それらの一つ以上のサブツリーとしてまとめて定義できます。一つのサブツリーには最大 200 ノードが定義できます。MIB オブジェクトを定義する手順を次に示します。

1. SNMP エージェントに追加したい MIB オブジェクトをリストアップする。
2. MIB オブジェクトの論理構成を決定する。  
MIB オブジェクトをグループとして論理的に構成します。例えば、`sysDescr`、`sysObjectID`、`sysUpTime`、`sysContact`、`sysName`、`sysLocation`、および `sysServices` の MIB-II オブジェクトは、すべて System グループに属します。`/var/opt/OV/share/snmp_mibs` ディレクトリの MIB モジュールに多くの例があるので参照してください。
3. 各サブツリーへのノードを定義する。  
各サブツリーにすべてのノードを定義します。ノードによっては、ほかのノードの子になることもあります。  
ノードを定義するとき、ASN.1 に定義されている次の規則に従ってください。
  - 任意数個の文字・数字・ハイフンを使用できる。
  - 英小文字で始める。
  - ハイフンで終わらせない。
  - ハイフンを連続して使用しない。
  - アンダースコアは使用しない。
 なお、次に示す規則もあります。
  - カウンタは文字「s」で終わる。
  - 各ノード名はユニークにする。
4. サブツリーの葉ノードを定義する。  
実在のオブジェクト（サブツリーの葉ノード）を定義します。  
実在のオブジェクトを定義するときは、ユニークな名称にしてください。オブジェクト名を決定するときの一般的な規則を次に示します。
  - 一つのグループに属するオブジェクト名は、すべてそのグループ名の接頭辞で始める。例えば、System グループのオブジェクトはすべて接頭辞 'sys' で始まる。
  - 接頭辞のあとの語を大文字で記述する。例えば、オブジェクト名 `sysContact` では `Contact` の C が大文字になる。
5. MIB ツリー内のオブジェクトの位置を決定する。  
MIB ツリーのどこにオブジェクトを置くかを決定します。  
`enterprises` サブツリーにある自分の会社名の下に作成した MIB を追加すればオブジェクト識別子は確実にユニークになります。  
独自の enterprise ID を指定する場合は、次の機関に自分の enterprise ID を登録する

と便利です。

Internet Assigned Numbers Authority  
URL: <http://www.iana.org/>  
Email: [iana-mib@iana.org](mailto:iana-mib@iana.org)

Internet Assigned Numbers Authority (IANA) に自分の enterprise ID を登録しておけば、自分自身の MIB を制御できるので、ほかの MIB との衝突を避けられます。

### 2.6.2 拡張 MIB 定義ファイルの作成

システムにルートユーザーとしてログインして、拡張 MIB 定義ファイル (`/etc/SnmpAgent.d/snmpd.extend`) に拡張 MIB オブジェクトを定義します。

`/etc/SnmpAgent.d/snmpd.extend` ファイルは、ユーザーが定義したオブジェクトから成る MIB モジュールで、SNMP エージェント上の MIB を拡張します。

`/etc/SnmpAgent.d/snmpd.extend` ファイルは、RFC1212:Concise MIB Definitions に定義されたマクロのテンプレートを使用して作成します。したがって、`/etc/SnmpAgent.d/snmpd.extend` ファイルを作成するときは、RFC1212 に記述された Abstract Syntax Notation One (ASN.1) フォーマットに従ってください。なお、インターネット標準 MIB-II、HP 企業固有 MIB および日立企業固有 MIB の MIB モジュールに関しては、以下のファイルを参照して下さい。

- `/var/opt/OV/share/snmp_mibs/eagent/rfc1213-MIB-II`
- `/var/opt/OV/share/snmp_mibs/eagent/rfc1285-FDDI`
- `/var/opt/OV/share/snmp_mibs/eagent/rfc1398-ETHER`
- `/var/opt/OV/share/snmp_mibs/eagent/hp-unix`
- `/var/opt/OV/share/snmp_mibs/eagent/hitachi-cometAgt`
- `/var/opt/OV/share/snmp_mibs/eagent/hitachi-cometAgt-aix`
- `/var/opt/OV/share/snmp_mibs/eagent/hitachi-cometAgt-solaris`

拡張 MIB オブジェクトには、非テーブル形式とテーブル形式があります。

#### 非テーブル形式

MIB オブジェクトに対してその値がユニークに決まります。

#### テーブル形式

複数の MIB オブジェクト列とエントリー行から成り、MIB オブジェクト列に対してユニークな ID を付けることでエントリーを識別します。したがって、一つの MIB オブジェクトに対してエントリー数分の MIB 値を持ちます。

次に、非テーブル形式とテーブル形式の拡張 MIB オブジェクトを `/etc/SnmpAgent.d/snmpd.extend` ファイルに作成する方法について説明します。

### (1) 非テーブル形式の拡張 MIB オブジェクト

使用できるマクロのテンプレートの形式を次に示します。下線部分のフィールドを記述する必要があります。

```

1. モジュール名 DEFINITION ::= BEGIN
2. --コメント
3. 企業名 OBJECT IDENTIFIER ::= {オブジェクト識別子}
4. ノード名 OBJECT IDENTIFIER ::= {オブジェクト識別子}
5. オブジェクト OBJECT-TYPE
6. SYNTAX データ型
7. ACCESS アクセス禁止レベル
8. STATUS 条件
   DESCRIPTION
9. "コメント"
10. READ-COMMAND: read_command
11. READ-COMMAND-TIMEOUT: コマンドの終了待ち時間
12. WRITE-COMMAND: write_command
13. WRITE-COMMAND-TIMEOUT: コマンドの終了待ち時間"
14. ::= {親ノード 識別番号}
   END

```

1. モジュール名  
ユーザーが作成した MIB モジュール名を記述します。
2. コメント  
コメントを追加するときは、そのコメントの前に二つのダッシュ (--) を記述します。
3. 企業名  
Internet Assigned Numbers Authority に登録した enterprise ID を記述します。  
オブジェクト識別子  
企業名に対応したオブジェクト識別子を記述します。例えば、日立の enterprise ID は hitachi であり、対応するオブジェクト識別子は {enterprises 116} です。
4. ノード名  
作成したい MIB オブジェクトのノード名を記述します。ノード名は複数記述でき、他ノードの子になることもできます。  
オブジェクト識別子  
ノード名に対応した親ノードの名称と作成したいノードのオブジェクト識別子を記述します。
5. オブジェクト  
オブジェクトのラベルを記述します。
6. データ型  
オブジェクトに対応したデータ型を記述します。SNMP エージェントで使用するデータ型を次の表に示します。

## 2. SNMP エージェントの環境設定

表 2-6 データ型

データ型	意味
INTEGER <sup>1</sup>	0 を含む正負の整数から成る簡易型。ラベルに 0 の値は使用できません。
OCTET STRING	0 以上のオクテットを取る簡易型。各オクテットは 8 ビット列です。
OBJECT IDENTIFIER	オブジェクト名の公認された記述型。 例：1.3.6.1.2.1.1.0
NULL	空の値。NULL はほとんど使用されません。
NetworkAddress	IP アドレスを表す型。
Counter <sup>2</sup>	変化を計算し最大値になるまで増加して最大値で 0 に戻る非負整数を表す型。最大値になって 0 に戻ると再び増加を始めます。
Counter64	最大値が $2^{64}-1$ である Counter 型
Gauge <sup>2</sup>	増加または減少しますが、最大値になっても 0 に戻らないで止まる非負整数を表す型。
TimeTicks <sup>2</sup>	ある時点からの時間を 1/100 秒単位で計算する非負整数を表す型。
Opaque	任意の符号化を表す型。
DisplayString	NVT ASCII 文字セットから取得できるテキスト情報を表す型。
PhysAddress	媒体アドレスを表す型。多くの媒体では 2 進表現になります。 例：インターネットアドレスは 6 オクテットのストリング

注 1

最大値は  $2^{31}-1$  です。

注 2

最大値は  $2^{32}-1$  です。

### 7. アクセス禁止レベル

許可されるアクセスのレベルを記述します。記述できる値を次に示します。

read-only

GetRequest は許可されるが SetRequest は許可されない。

read-write

GetRequest と SetRequest の両方が許可される。

### 8. 条件

実装で要求される条件を記述します。mandatory, optional, obsolete, または deprecated が記述できます。通常は mandatory を記述します。

### 9. コメント

オブジェクトに関するコメントを記述します。各ラベルのあとにオブジェクトに関連する情報を記述します。

## 10. read\_command

GetRequest または GetNextRequest を受け取ったときに、SNMP エージェントが実行するコマンドを記述します。このコマンドは READ-COMMAND ラベルに記述します。read\_command は完全パス名で記述してください。ACCESS の記述が read-only または read-write のときに記述します。結果は標準出力または標準エラー出力に出力してください。

## 11. コマンドの終了待ち時間 ~ 1 ~ 2 けたの数字 ((1 ~ 90)) 《3》

SNMP エージェントが read\_command で記述したコマンドの終了を待つ時間を秒単位で記述します。この値は READ-COMMAND-TIMEOUT ラベルに記述します。コマンドが記述した時間までに終了しない場合、SNMP エージェントはコマンドを強制終了させ、マネージャーに次のようにエラー応答します。

SNMPv1 の場合：noSuchName

SNMPv2c の場合：noSuchInstance

この記述は省略できます。省略すると 3 秒が仮定されます。

## 12. write\_command

SetRequest を受け取ったときに、SNMP エージェントが実行するコマンドを記述します。このコマンドは WRITE-COMMAND ラベルで記述します。write\_command は完全パス名で記述してください。ACCESS の記述が read-write のときに記述しません。

## 13. コマンドの終了待ち時間 ~ 1 ~ 2 けたの数字 ((-1, 1 ~ 90)) 《3》

SNMP エージェントが write\_command で記述したコマンドの終了を待つ時間を秒単位で記述します。この値は WRITE-COMMAND-TIMEOUT ラベルに記述します。-1 を記述すると、SNMP エージェントは write\_command で記述したコマンドの終了を待たないで応答します。コマンドが記述した時間までに終了しない場合、SNMP エージェントはコマンドを強制終了させ、マネージャーに次のようにエラー応答します。

SNMPv1 の場合：genErr

SNMPv2c の場合：commitFailed

この記述は省略できます。省略すると 3 秒が仮定されます。

SNMP エージェントはコマンドを一つずつ処理し、応答を待ってから次のコマンドを処理します。なお、アクセス禁止レベルおよびファイルの読み込み・書き込み許可に関連はありません。

## 14. 親ノード

親になるノード名を記述します。このノード名はすでに定義されているノード名である必要があります。

## 識別番号

識別番号には、親ノードに付けたオブジェクトをユニークに識別する番号を記述します。

## (2) テーブル形式の拡張 MIB オブジェクト

SNMP テーブルは SYNTAX SEQUENCE OF と INDEX の条項を使用して定義します。使用できるマクロのテンプレートの形式を次に示します。下線部分のフィールドを記述する必要があります。この形式は、オブジェクト 1 とオブジェクト 2 から成るテーブル形式の MIB オブジェクトをモデルにしています。各フィールドの記述について説明がないものは、非テーブル形式の拡張 MIB オブジェクトの定義と同じです。

```

モジュール名 DEFINITIONS ::= BEGIN
--コメント
企業名          OBJECT IDENTIFIER ::= {オブジェクト識別子}
ノード名        OBJECT IDENTIFIER ::= {オブジェクト識別子}
1. テーブル名 OBJECT-TYPE
2.   SYNTAX SEQUENCE OF エントリー・データ型
   ACCESS not-accessible
   STATUS mandatory
   DESCRIPTION
   "コメント"
3.   FILE-COMMAND:file_command
4.   FILE-COMMAND-FREQUENCY:ファイルコマンドの実行間隔
5.   PIPE-IN-NAME:pipe_in_name
6.   PIPE-OUT-NAME:pipe_out_name
7.   PIPE-FREQUENCY:PIPE書き込み間隔
8.   APPEND-COMMUNITY-NAME:{true|false}
9.   FILE-NAME:file_name"
   ::= {親ノード 識別番号}
10. エントリー名 OBJECT-TYPE
   SYNTAX エントリー・データ型
   ACCESS not-accessible
   STATUS mandatory
   DESCRIPTION
   "コメント"
   INDEX {オブジェクト1}
   ::= {テーブル名 1}
   エントリー・データ型 ::=
   SEQUENCE {
     オブジェクト1 データ型1,
     オブジェクト2 データ型2
   }
11. オブジェクト1 OBJECT-TYPE
   SYNTAX データ型1
   ACCESS アクセス禁止レベル
   STATUS 条件
   DESCRIPTION
   "コメント"
   ::= {エン트리名 1}
   オブジェクト2 OBJECT-TYPE
   SYNTAX データ型2
   ACCESS アクセス禁止レベル
   STATUS 条件
   DESCRIPTION
   "コメント"
   ::= {エン트리名 2}
END

```

## 1. テーブル名

MIB テーブルのラベルを記述します。

## 2. エントリー・データ型

MIB テーブルエントリーのデータ型を記述します。通常、MIB テーブルエントリーのラベルの先頭文字を大文字にした名称を用います。エントリー・データ型は、MIB テーブルの列要素を表しています。この例では、MIB テーブルの列要素としてオブ

## 2. SNMP エージェントの環境設定

ジェクト 1 とオブジェクト 2 を記述しています。

### 3. file\_command

SNMP エージェントが GetRequest, GetNextRequest, または SetRequest を受け取ったときに, FILE-NAME ラベルに記述した file\_name を読む前に実行するコマンドを記述します。SetRequest を受け取ったときは, file\_name の読み出しの前後で実行されます。このコマンドは FILE-COMMAND ラベルに記述します。

file\_command は完全パス名で記述してください。

コマンドの応答が返ってくるまでの監視時間を, extsubagt のオプション (-fcmdguard) に 90 秒以内で設定できます。省略すると, 10 秒が仮定されます。監視時間内に, 応答が返ってこない場合, SNMP エージェントはこのコマンドを kill し, 次のように動作します。

SNMPv1 の場合

マネージャーに genErr をエラー応答

SNMPv2c の場合

get\_request のとき: マネージャーに noSuchName をエラー応答

get\_next\_request のとき: マネージャーに EndOfMibView をエラー応答

set\_request のとき: マネージャーに genErr をエラー応答

注 Varbind の値。ステータスは正常。

extsubagt プロセスのオプションを, 起動時, または snmpstart コマンド実行時に, 常に有効になるように設定できます。extsubagt プロセスのオプションの設定については, 「3.1.1 SNMP エージェントの起動」を参照してください。

### 4. ファイルコマンドの実行間隔 ~ 数字 ((0 ~ 2147483647)) 《10》

SNMP エージェントが GetRequest または GetNextRequest を受け取ったときに, file\_command を最後に実行してからの経過時間がファイルコマンドの実行間隔以内の場合には, コマンドを実行しません。この値は秒単位で, FILE-COMMAND-FREQUENCY ラベルに記述します。SetRequest を受け取ったときは, file\_command が最後に実行されてからの時間をチェックしません。この記述は FILE-COMMAND ラベルが記述されたときだけ有効です。また, この記述は省略できます。省略すると 10 秒が仮定されます。

### 5. pipe\_in\_name

PIPE-OUT-NAME ラベルに記述したファイルにデータを書き込んだあとに, UNIX プロセスが SNMP エージェントに処理の完了を通知するためのファイルを記述します。このファイルは PIPE-IN-NAME ラベルに記述します。pipe\_in\_name は完全パス名で記述してください。

pipe\_out\_name にデータを書き込んでから処理結果が書き込まれるまでの監視時間を, extsubagt のオプション (-pipeguard) に 90 秒以内で設定できます。省略すると, 20 秒が仮定されます。

また, 監視時間の長さによっては, 処理結果が書き込まれる前に, 次の要求が書き込

まれる場合があります。このような場合でも、要求と応答を一致させるために、`pipe_in_name` に書き込むデータの先頭に識別番号を付けることができます。識別番号も、`extsubagt` コマンドのオプション (`-invokeid`) で指定します。

`pipe_in_name` に渡すデータを次の表に示します。

表 2-7 `pipe_in_name` に渡すデータ

データ項目	渡される値
識別番号	どの要求に対する処理結果かを識別する番号です。 <code>extsubagt</code> のオプション ( <code>-invokeid</code> ) が指定されたときだけ、付けられません。形式は、 <code>xxxxxxxxx.yyyyyyyy</code> ( <code>xxxxxxxxx</code> : 通算秒, <code>yyyyyyyyy</code> : マイクロ秒) です。
結果コード	0

SNMP エージェントは、`pipe_in_name` の内容が 0 以外 (4 バイトの数値) のとき、または監視時間内に `pipe_in_name` に処理結果が書き込まれないときは、次のように動作します。

#### SNMPv1 の場合

マネージャーに `genErr` をエラー応答

#### SNMPv2c の場合

`get_request` のとき：マネージャーに `noSuchName` をエラー応答

`get_next_request` のとき：マネージャーに `EndOfMibView` をエラー応答

`set_request` のとき：マネージャーに `genErr` をエラー応答

注 `Varbind` の値。ステータスは正常。

`extsubagt` プロセスのオプションを、起動時または `snmpstart` コマンド実行時に、常に有効になるように設定できます。`extsubagt` プロセスのオプションの設定については、「3.1.1 SNMP エージェントの起動」を参照してください。

#### 6. `pipe_out_name`

SNMP エージェントが `GetRequest`、`GetNextRequest` または `SetRequest` を受け取ったときに、`FILE-NAME` ラベルに記述した `file_name` を読む前に、任意の UNIX プロセスにデータを渡すために書き込むファイルに記述します。`SetRequest` を受け取ったときは、`file_name` の読み出しの前で実行されます。このファイルは `PIPE-OUT-NAME` ラベルに記述します。`pipe_out_name` は完全パス名で記述してください。この記述は省略できます。ただし、`PIPE-OUT-NAME` ラベルは `PIPE-IN-NAME` ラベルと対で記述します。また、`pipe_in_name` と同様に、`pipe_out_name` に書き込むデータの先頭に、識別番号を付けることができます。識別番号は `extsubagt` プロセスのオプション (`-invokeid`) で指定します。

`pipe_out_name` に渡すデータを次の表に示します。データとデータは、1 個の空白で区切られています。データの形式は文字列です。データの終端には `"¥0"` が追加されています。

表 2-8 pipe\_out\_name に渡すデータ

データ項目	渡される値
識別番号	どの要求かを識別するための番号。extsubagt のオプション (-invokeid) が指定されたときだけ付けられます。形式は、xxxxxxxxx.yyyyyyy (xxxxxxxxx : 通算秒, yyyyyyy : マイクロ秒) です。
マネージャーの IP アドレス	マネージャーの IP アドレス。インターネットのドット記法で表現されます。
コミュニティ名	リクエスト中のコミュニティ名。
オブジェクト識別子	リクエスト中のオブジェクト識別子。ドット記法で表現されます。
オブジェクトのシンタクス	リクエスト中のオブジェクトのシンタクス。次のどれかの値が渡されます。 Integer, OctetString, ObjectIdentifier, Null, NetworkAddress, IPAddress, Counter, Counter64, Gauge, TimeTicks, Opaque, DisplayString, PhysAddress
PDU タイプ	リクエストの種類。次のどれかの値が渡されます。 GetRequest, GetNextRequest, SetRequest, PostSetRequest PostSetRequest は、file_name の読み出したあとに渡される値です。
インスタンス名	リクエスト中の接尾辞。
Set 値	SetRequest の場合、SetRequest 中の Set 値が渡されます。

extsubagt プロセスのオプションを、起動時または snmpstart コマンド実行時に、常に有効になるように設定できます。プロセスのオプションの設定については、「3.1.1 SNMP エージェントの起動」を参照してください。

#### 7. PIPE 書き込み間隔 ~ 数字 ((0 ~ 2147483647)) 《10》

SNMP エージェントが GetRequest または GetNextRequest を受け取ったときに、pipe\_out\_name に最後にデータを書き込んでからの経過時間が PIPE 書き込み間隔以内であれば、pipe\_out\_name にデータを書き込みません。この値は、秒単位で PIPE-FREQUENCY ラベルに記述します。

SetRequest を受け取ったときは、pipe\_out\_name に最後に書き込んでからの時間をチェックしません。この記述は、PIPE-IN-NAME ラベルと PIPE-OUT-NAME ラベルの両方が記述されたときにだけ有効です。省略すると 10 秒が仮定されます。

#### 8. APPEND-COMMUNITY-NAME :{ true | false }

true を記述すると、SNMP エージェントがリクエスト (GetRequest, GetNextRequest または SetRequest) を受け取ったときに読み書きするファイルとして、FILE-NAME ラベルに記述した file\_name に、リクエスト中に含まれるコミュニティ名を付けます。false を記述したときは、file\_name にコミュニティ名を付けません。この記述はラベルごと省略できます。省略すると false が仮定されます。

## 9. file\_name

SNMP エージェントが GetRequest, GetNextRequest または SetRequest を受け取ったときに、読み書きするファイルを記述します。このファイルは FILE-NAME ラベルに記述します。file\_name は完全パス名で記述してください。この記述は省略できません。

SNMP エージェントは、file-command を実行する前、または pipe\_out\_name にデータを書き込む前にファイルが存在しているかどうかをチェックします。ファイルが存在しない場合はエラーとなり、MIB 値を取得できません。ファイルのチェックは、ファイルが存在しているかどうかだけをチェックしています。ファイルの内容はチェックしていません。

## 10. エントリー名

MIB テーブルのエントリーのラベルを記述します。このオブジェクトのシンタクスは、エントリー・データ型で定義されます。このオブジェクト定義の INDEX ラベルには、MIB テーブル列の中でエントリー行をユニークに識別できる MIB 値を持つ MIB オブジェクトを記述します。この例では、オブジェクト 1 を記述しています。

## 11. オブジェクト 1

以降は、MIB テーブルの列要素の MIB オブジェクト定義について記述します。

## (3) 注意事項

/etc/SnmpAgent.d/snmpd.extend ファイルは次に示す点で RFC とは異なるため、注意する必要があります。

- /etc/SnmpAgent.d/snmpd.extend ファイルでは、imports と exports の条項は不要なため、記述しても無視されます。
- DESCRIPTION フィールドは必ず記述してください。このフィールドで実行したいコマンドなどを定義してください。コマンドの説明を DESCRIPTION フィールドに追加しておけば、マネージャーの要求に対して実行しているコマンドの内容がわかるので便利です。
- DESCRIPTION フィールド中の各ラベルを 2 個以上重複して記述すると、2 個目以降は 1 個目のラベルの値として解釈されます。
- DESCRIPTION フィールド中の各ラベルは、定義マクロのテンプレートの形式で示した順序で記述してください。誤った順序で記述すると、定義文解析エラーになります。
- 拡張 MIB オブジェクト定義機能を使用してテーブル形式の拡張 MIB オブジェクトを定義する場合、テーブルの列数は 255 列以下で定義してください。
- オブジェクト名およびエントリー名は 59 文字以下で定義してください。

## 2.6.3 SNMP 要求時に実行するコマンドの作成

ここでは、/etc/SnmpAgent.d/snmpd.extend ファイルに記述するユーザー独自のシェルコマンドの作成方法について説明します。

## 2. SNMP エージェントの環境設定

シェルコマンドは UNIX のシェルスクリプトまたはプログラムです。/opt/OV/prg\_samples/eagent ディレクトリに、/etc/SnmpAgent.d/snmpd.extend ファイルの作成例を提供しています。作成例にシェルコマンドのサンプルが記載されています。

コマンド名を記述するときの注意事項を次に示します。

- コマンド名は /etc/SnmpAgent.d/snmpd.extend ファイルの DESCRIPTION フィールドに記述してください。
- コマンド名の最大長は 5,120 字です。
- コマンド名は複数行に継続できます。継続行は ¥ で終了させてください。
- READ-COMMAND, WRITE-COMMAND, FILE-COMMAND で定義したコマンドは root 権限で実行します。これらのコマンドに root が実行可能なファイル属性を設定してください。

コマンドの作成手順を次に示します。

1. コマンドを実行したいシステムへログインする。
2. スクリプトまたはプログラムを記述する。
3. シェルコマンドの動作を確認する。
4. 出口コードをチェックする。
5. シェルコマンドの引数を確認する。

次に、コマンドの作成方法を説明します。

### (1) コマンドを実行したいシステムへのログイン

コマンドを実行したいシステムへログインします。

### (2) スクリプトまたはプログラムの記述

スクリプトまたはプログラムの記述方法を次に示します。

#### 引数

SNMP エージェントがコマンドに引数を渡すには、次の表に示す引数を任意で使用します。記述は順不同です。

表 2-9 引数

引数	渡される値
\$i	マネージャーの IP アドレス。インターネットのドット記法で表現されます。
\$c	リクエスト中のコミュニティ名。
\$o	リクエスト中のオブジェクト識別子。ドット記法で表現されます。
\$s	オブジェクトのシンタックス。次のどれかの値が渡されます。 Integer, OctetString, ObjectIdentifier, Null, NetworkAddress, IpAddress, Counter, Counter64, Gauge, TimeTicks, Opaque, DisplayString, PhysAddress

引数	渡される値
\$r	リクエストの種類。次のどれかの値が渡されます。 GetRequest, GetNextRequest, SetRequest, PostSetRequest PostSetRequest は、file_name の読み出したあとにだけ渡される値です。
\$I	リクエスト中の接尾辞。
\$ *	すべての引数, \$i \$c \$o \$s \$r \$I と指定したのと同じです。
\$\$	\$ 文字を引数として渡したい場合, \$\$ と指定します。\$ の 1 文字が引数として渡され ます。
Set 値	SetRequest, および PostSetRequest の場合, SetRequest 中の Set 値が渡されます。 この値は引数リストの最後に渡されます。

非テーブル形式の MIB オブジェクトで、`/opt/OV/prg_samples/eagent/num_widgets` コマンドを実行するとき、SNMP エージェントがコマンドにマネージャーの IP アドレス、コミュニティ名、およびオブジェクト識別子の値を渡す例を示します。`/etc/SnmpAgent.d/snmpd.extend` ファイルの DESCRIPTION フィールドは次のように記述します。

```
READ-COMMAND: /opt/OV/prg_samples/eagent/num_widgets $i $c $o
```

また、テーブル形式の MIB オブジェクトで、FILE-NAME ラベルに記述したファイルを読む前に `/opt/OV/prg_samples/eagent/update_inetd` コマンドを実行するとき、SNMP エージェントがコマンドにリクエストの種類を渡す例を示します。`/etc/SnmpAgent.d/snmpd.extend` ファイルの DESCRIPTION フィールドは次のように記述します。

```
FILE-COMMAND: /opt/OV/prg_samples/eagent/update_inetd $r
```

引数の記述がない場合、SNMP エージェントは READ-COMMAND ラベルおよび FILE-COMMAND ラベルに記述したコマンドに引数は渡しません。

SNMP エージェントは、WRITE-COMMAND ラベルおよび FILE-COMMAND ラベルに記述したコマンドに引数の記述がない場合、リクエスト中の Set 値を一つだけ渡します。

get および set オペレーションで同じコマンドを使用する場合の引数

READ-COMMAND ラベルおよび WRITE-COMMAND ラベルに記述する二つのコマンドに同じコマンドを使用したい場合は、コマンドに渡す引数で GetRequest か SetRequest かを区別します。記述する引数を次に示します。

READ.REQ

get オペレーションで使用したい場合

WRITE.REQ

set オペレーションで使用したい場合

## 2. SNMP エージェントの環境設定

/usr/bin/my\_command というコマンドを使用する場合は次のように記述します。

### READ-COMMAND

```
/usr/bin/my_command READ.REQ
```

### WRITE-COMMAND

```
/use/bin/my_command WRITE.REQ
```

### サーチパス

コマンド名は、完全パス名で記述してください。

### 戻り値

READ-COMMAND ラベルに記述したコマンドの戻り値は、標準出力または標準エラー出力に出力してください。

### 実行

作成したコマンドは、/bin/sh によって実行されたように見えます。exit, read, if, for のシェルコマンドも記述できます。

### 出口コード

READ-COMMAND ラベルおよび WRITE-COMMAND ラベルに記述したシェルコマンドは必ず 0 で終了してください。0 以外の出口コードで終了すると、次のように動作します。

### SNMPv1 の場合

get-request のとき：マネージャーに noSuchName をエラー応答

get-next-request のとき：次のオブジェクトを検索

set-request のとき：マネージャーに genErr をエラー応答

### SNMPv2c の場合

get-request のとき：マネージャーに noSuchInstance をエラー応答

get-next-request のとき：次のオブジェクトを検索

set-request のとき：マネージャーに commitFailed をエラー応答

FILE-COMMAND ラベルに記述したコマンドの出口コードは、必ず 0 で終了してください。0 以外の出口コードで終了すると、次のように動作します。

### SNMPv1 の場合

get-request のとき：マネージャーに noSuchName をエラー応答

get-next-request のとき：次のオブジェクトを検索

set-request のとき：マネージャーに genErr をエラー応答

### SNMPv2c の場合

get-request のとき：マネージャーに noSuchInstance をエラー応答

get-next-request のとき：次のオブジェクトを検索

set-request で非テーブル型で SetRequest/PostSetRequest のとき：マネージャーに commitFailed をエラー応答

set-request でテーブル型で SetRequest のとき：マネージャーに genErr をエ

ラー応答

set-request でテーブル型で PostSetRequest のとき：マネージャーに  
commitFailed をエラー応答

また，SNMP エージェントがユーザーのシェルコマンドを実行した結果，コマンドがないなどのエラーが発生した場合も，すべて同じ動作をします。

### (3) シェルコマンドの動作確認

コマンドを実行して，シェルコマンドが正常に実行されることを確認します。

/etc/SnmpAgent.d/snmpd.extend ファイル中にある fmailListMsgs 関連のコマンドが正常に実行できることを確かめるためには，次に示すコマンドを実行します。

```
/usr/bin/mailq
```

正常に実行した場合，コマンドはメールメッセージのリストを標準出力します。

### (4) 出口コードのチェック

次に示すコマンドを実行して，値が 0 ならば，シェルコマンドは成功したことを示します。

```
echo $?
```

### (5) シェルコマンドの引数の確認

シェルコマンドで引数を使用している場合は，その引数を確認します。

/etc/SnmpAgent.d/snmpd.extend ファイルの定義および引数の確認の例を次に示します。

/etc/SnmpAgent.d/snmpd.extend ファイルの定義

```
READ-COMMAND:/opt/OV/prg_samples/eagent/num_widgets $i $c $o $s
```

引数の確認

```
num_widgets 15.2.3.149 public 1.3.6.4.4242.3.1 Gauge
```

## 2.6.4 SNMP 要求時に処理するファイルの作成

ここでは，テーブル形式の拡張 MIB オブジェクトを定義した場合の，FILE-NAME ラベルに記述するファイルの作成方法について説明します。

ファイル名を記述するときの注意事項を次に示します。

- ファイル名は /etc/SnmpAgent.d/snmpd.extend ファイルの DESCRIPTION フィールドに記述してください。
- ファイル名は完全パス名で記述してください。

## 2. SNMP エージェントの環境設定

- ファイル名の最大長は 5,120 字です。
- ファイル名は複数行に継続できます。継続行は ¥ で終了させてください。

次に、ファイルの内容の記述方法について説明します。

1. 各テーブル行は、1 行で終わらせる必要があります。ただし、行の終わりに ¥ 文字を記述することで継続行とすることができます。こうすることで、各テーブル行を複数行にわたって記述できます。

例えば、ファイルの内容が次のような場合、SNMP エージェントは、1 行と 5 列の要素から成るテーブルとみなします。

```
Column1 "Column # 2" ¥  
        "Column # 3" Column4 Column5
```

2. 各行の列要素は、空白で区切られるか、または " で囲む必要があります。" で囲まれた列要素の中に " や ¥ を含む場合、次のように記述します。

```
"        ¥"  
¥        ¥¥
```

例えば、ファイルの内容が次のような場合、

```
"This is an ¥"example¥" of a column with ¥¥" style quotes"
```

SNMP エージェントは次のように解釈します。

```
This is an "example" of a column with ¥" style quotes
```

3. MIB オブジェクトの SYNTAX が PhysAddress, OCTET STRING, Opaque の場合に、文字列の先頭が 0x で始まっていれば 16 進数と解釈します。例えば、ファイルの内容が 0x0800093519D0 の場合、SNMP エージェントは 16 進数として (0800093519D0)<sub>16</sub> をマネージャーに応答します。
4. 第 1 カラムが # の場合、コメントとして読み飛ばします。

### 2.6.5 サブエージェントの再構成

SNMP エージェントは、起動時に /etc/SnmpAgent.d/snmpd.extend の内容をチェックします。内容にエラーがなければ、オブジェクトを追加します。

/etc/SnmpAgent.d/snmpd.extend にシンタクスエラーがないことを確認してから /opt/CM2/ESA/bin/snmpstart コマンドを使用して SNMP エージェントを起動してください。/etc/SnmpAgent.d/snmpd.extend のシンタクスチェックは -p -apall オプションを指定した extsubagt プロセスで実行できます。なお、extsubagt プロセスは、snmpstop コマンドを使用して SNMP エージェントを停止した状態で実行してください。

## 2.6.6 マネージャーのコマンドによるオブジェクトの確認

追加したオブジェクトに応答することを、マネージャーのコマンドによるオブジェクトの確認が提供するコマンドで確認します。コマンドについては、各マネージャーのコマンドによるオブジェクトの確認のドキュメントを参照してください。

## 2.6.7 すべての SNMP エージェントの設定

次のどちらかの方法で、SNMP エージェントを設定してください。

- すべての SNMP エージェントに `/etc/SnmpAgent.d/snmpd.extend` をコピーする方法  
コピーしたあと、SNMP エージェントを再構成してください。
- SNMP エージェントごとに `/etc/SnmpAgent.d/snmpd.extend` ファイルを作成する方法  
異なる SNMP エージェントで異なるオブジェクトを管理したい場合、各 SNMP エージェントに別の `/etc/SnmpAgent.d/snmpd.extend` ファイルを作成できます。この場合、使用するオブジェクト識別子を必ずユニークにしてください。各オブジェクト識別子の関連記述は、すべての SNMP エージェントで同じにしてください。図 2-4 の手順を繰り返して、各 SNMP エージェントの `/etc/SnmpAgent.d/snmpd.extend` ファイルを作成します。

## 2.6.8 拡張 MIB オブジェクトのマネージャーへのコピー

SNMP エージェントに追加した新しい MIB オブジェクトにマネージャーがアクセスできるように、`/etc/SnmpAgent.d/snmpd.extend` ファイルをマネージャーにコピーしておく必要があります。

NNM の場合、デフォルトの MIB モジュールのディレクトリにコピーしておくこと、MIB モジュールファイルの参照が簡単にできます。デフォルトのディレクトリは `/var/opt/OV/share/snmp_mibs` です。

## 2.6.9 マネージャーへの MIB の統合

MIB モジュールをマネージャーにコピーしたら、その新しいオブジェクトをマネージャーの MIB に統合します。MIB を統合する手順については、各マネージャーが提供するドキュメントを参照してください。NNM および SubManager の場合の手順を、例として次に示します。

1. 新しい MIB をマネージャーの MIB にロードする。  
NNM および SubManager が新しい MIB のオブジェクトにアクセスするには、これらのオブジェクトを定義する MIB モジュールが NNM および SubManager の MIB にロードされている必要があります。この操作には、NNM では `xnmloadmib` コマンド、または `[MIB のロード/アンロード: SNMP ...]` を使用します。SubManager

## 2. SNMP エージェントの環境設定

では `xnmloadmib` コマンドを使用します。

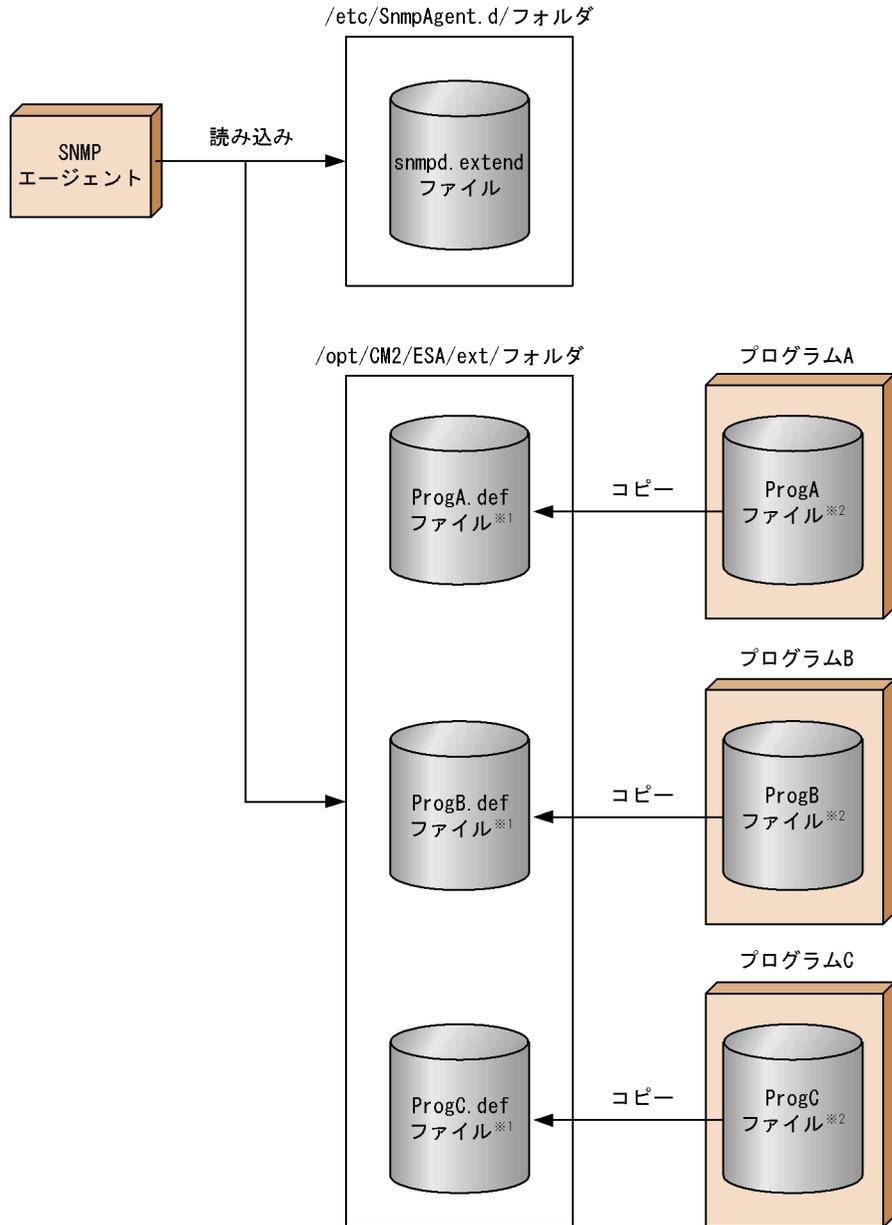
2. 新しいオブジェクトの管理には、NNM では [ MIB のブラウズ : SNMP ... ], [ データ収集としきい値 : SNMP ... ], [ MIB アプリケーションビルダ : SNMP ... ] および [ MIB アプリケーションビルダ : SNMP ... ] で作成したアプリケーションを使用する。

### 2.6.10 複数の拡張 MIB 定義ファイルの設定

複数の拡張 MIB 定義ファイルを SNMP エージェントに追加する場合、新規に拡張 MIB 定義ファイルを作成します。

プログラム単位で拡張 MIB 定義ファイルを作成し、複数の拡張 MIB 定義ファイルを SNMP エージェントに設定する流れを次の図に示します。

図 2-5 プログラム単位に拡張 MIB 定義ファイルを作成する流れ



注※1 拡張MIB定義ファイルの拡張子を.defに変更してコピーしたファイルです。

注※2 プログラム単位に作成した拡張MIB定義ファイルです。

### (1) 拡張 MIB 定義ファイルの追加方法

作成した拡張 MIB 定義ファイルを SNMP エージェントに追加する方法について説明し

## 2. SNMP エージェントの環境設定

ます。

### (a) 拡張 MIB 定義ファイルの格納

/opt/CM2/ESA/ext ディレクトリ配下に、拡張 MIB 定義ファイルの拡張子を .def に変更したファイルをコピーするか、またはシンボリックリンクを作成します。

作成するファイル名の長さは、拡張子を含めて 12 文字以内（8 文字以内 . 拡張子）です。

作成するファイルの属性は、extsubagt プロセス（所有者：bin，グループ：bin）で読み込めるように設定してください。

追加する拡張 MIB オブジェクトに起動オプションを設定する場合は、/opt/CM2/ESA/ext ディレクトリ配下にオプション定義ファイルを作成します。オプション定義ファイルの作成方法については、「(3) 拡張 MIB オブジェクトの起動オプション定義ファイルの設定方法」を参照してください。

### (b) 拡張 MIB オブジェクトの読み込み

拡張 MIB オブジェクトを提供するために、拡張 MIB 定義ファイルに指定された extsubagt プロセスを起動する必要があります。

拡張 MIB 定義ファイルに指定された extsubagt プロセスを起動する方法には、SNMP エージェントを停止して実行する方法と停止しないで実行する方法があります。それぞれの手順を次に示します。

SNMP エージェントを停止して拡張 MIB 定義ファイルを追加する場合

SNMP エージェントを停止して拡張 MIB 定義ファイルを追加する手順を次に示します。

1. /opt/CM2/ESA/bin/snmpstop コマンドをスーパーユーザーで実行する。  
SNMP エージェントが停止されます。
2. /opt/CM2/ESA/bin/snmpstart コマンドをスーパーユーザーで実行する。  
SNMP エージェントが起動され、拡張 MIB 定義ファイルに指定された extsubagt プロセスが起動します。

SNMP エージェントを停止しないで拡張 MIB 定義ファイルを追加する場合

1. /opt/CM2/ESA/bin/snmpstart -e コマンドをスーパーユーザーで実行する。  
拡張 MIB 定義ファイルに指定された extsubagt プロセスが起動します。

## (2) 拡張 MIB 定義ファイルの起動の確認

定義した拡張 MIB 定義ファイルに指定された extsubagt プロセスが起動されたかどうかは、snmpcheck コマンドを実行して確認します。

/etc/SnmpAgent.d/snmpd.extend ファイル、および /opt/CM2/ESA/ext ディレクトリ配下に拡張 MIB 定義ファイル（ProgA.def，ProgB.def，ProgC.def）を設定した場合を例

に、すべて正常に起動しているかどうかを確認する方法を次の図に示します。

図 2-6 拡張 MIB 定義ファイルの起動の確認方法

```
#/opt/CM2/ESA/bin/snmpcheck
snmpdm    running pid=29771
mib2agt   running pid=29779
hp_unixagt    running pid=29787
trapdestagt    running pid=29789
extsubagt     running pid=29795  file = /etc/SnmpAgent.d/
snmpd.extend
extsubagt     running pid=3593  file = /opt/CM2/ESA/ext/ProgA.def
extsubagt     running pid=3594  file = /opt/CM2/ESA/ext/ProgB.def
extsubagt     running pid=3596  file = /opt/CM2/ESA/ext/ProgC.def
htc_unixagt1  running pid=29791
htc_unixagt2  running pid=29793
```

### (3) 拡張 MIB オブジェクトの起動オプション定義ファイルの設定方法

拡張 MIB オブジェクトに起動オプションを指定する場合の手順を次に示します。このオプションは OS の起動時または SNMP エージェントの起動時に有効となります。

1. /opt/CM2/ESA/ext ディレクトリ配下に起動オプション定義ファイルの拡張子を opt として作成する。  
作成するファイルの属性は、extsubagt プロセス（所有者：bin，グループ：bin）で読み込めるように設定してください。  
作成する拡張 MIB 定義ファイル名および起動オプション定義ファイル名を次に示します。
  - 拡張 MIB 定義ファイル名：/opt/CM2/ESA/ext/ProgA.def
  - 起動オプション定義ファイル名：/opt/CM2/ESA/ext/ProgA.opt
2. 拡張 MIB オブジェクトの実行時に有効にしたいオプションを設定する。  
次に示す設定内容でオプションを指定する場合を例にして、オプションの設定例を次に示します。
  - FILE\_COMMAND で指定したコマンドの応答時間：20 秒
  - PIPE\_IN\_NAME および PIPE\_OUT\_NAME で指定したパイプの応答監視時間：25 秒
  - パイプによるデータの送受信データの一致判定に識別番号の使用有無：使用する

図 2-7 拡張 MIB オブジェクトの実行時に有効にしたいオプションの設定例

```
SNMP_EXTAGT_OPTIONS="-fcmguard 20 -pipeguard 25 -invokeid"
export SNMP_EXTAGT_OPTIONS
```

## 2. SNMP エージェントの環境設定

SNMP エージェントを停止しないで、起動中の拡張 MIB オブジェクトのオプションを変更する場合は、次の手順を実行します。

1. snmpcheck コマンドを実行する。
2. オプションを変更したい拡張 MIB オブジェクトのプロセス番号を確認する。
3. スーパーユーザーで「kill -9 手順 2 で確認したプロセス番号」を実行する。  
拡張 MIB オブジェクトが停止されます。
4. 起動オプション定義ファイルを変更する。
5. /opt/CM2/ESA/bin/snmpstart -e をスーパーユーザーで実行する。  
SNMP エージェントが起動されます。

### (4) 注意事項

- /opt/CM2/ESA/ext ディレクトリ配下にあるファイルで、拡張子が .def 以外のファイルは、拡張 MIB 定義ファイルとして読み込まれません。
- SNMP エージェントの起動時に、拡張 MIB 定義ファイルの内容はチェックされ、エラーがあれば起動されないため、起動前にシンタクスチェックしておく必要があります。シンタクスチェックは、次のプロセスで実行できます。

HP-UX (IPF) の場合

```
/opt/CM2/ESA/bin/extsubagt -e 拡張 MIB 定義ファイル -p -apall
```

Solaris , AIX , Linux , Linux (IPF) の場合

```
/usr/sbin/extsubagt -e 拡張 MIB 定義ファイル -p -apall
```

- 拡張 MIB 定義ファイルの定義数は、最大 100 個です。

## 2.6.11 拡張 MIB オブジェクトの定義例

拡張 MIB オブジェクトの定義例を次の順で説明します。

- MIB オブジェクトの定義
- システム (flintagent) へのログイン
- /etc/SnmpAgent.d/snmpd.extend ファイルの作成例

### (1) MIB オブジェクトの定義

ユーザーが Flintstones Company の社員の状態で、次に示す目的で MIB オブジェクトを記述します。

1. システムを使用しているユーザーをリストアップする。
2. 各マシンのメモリーの内容を参照する。
3. メールキューを管理する。
4. 無人のシステムで 1 時間あたりに発生するウィジェット数を管理する。
5. LP スケジューラを見失わないようにする。

6. デフォルトのプリンタを参照する。
7. ユーザー ID, ユーザーが使用しているディスクスペース, ユーザーの email のアドレスリストを参照する。
8. ルートプロセスの一覧を出力する。
9. inetd ( 1M ) の構成ファイルの内容を変更する。

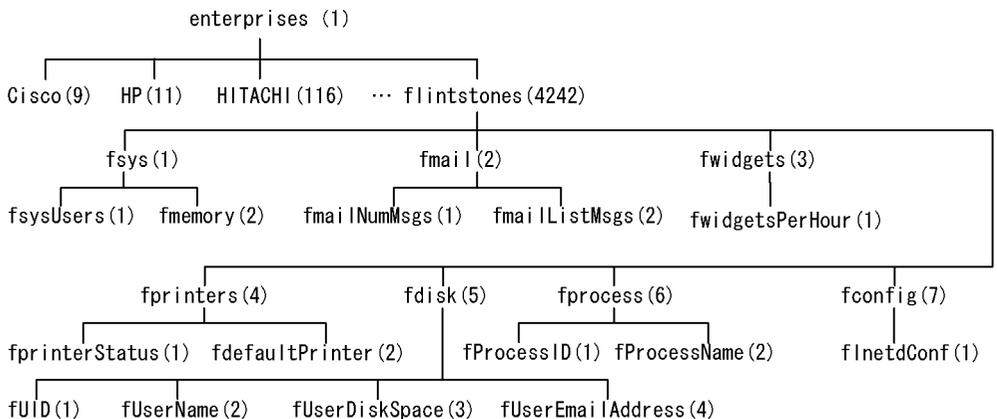
上記の MIB オブジェクトの定義については、「(3) /etc/SnmpAgent.d/snmpd.extend ファイルの作成例」の MIB オブジェクト定義の説明を参照してください。

エージェントシステムは flintagent, デフォルトのコミュニティ名は public, set コミュニティ名は secret です。オブジェクト識別子がユニークになるように, MIB オブジェクトは flintstones ( 4242 ) サブツリーの中に定義することにします。MIB ツリーの階層を次に示します。

```
iso (1) ::= { 1 }
org (3) ::= { 3 }
dod (6) ::= { 6 }
internet (1) OBJECT IDENTIFIER ::= { dod 1 }
private (4) OBJECT IDENTIFIER ::= { internet 4 }
enterprises (1) OBJECT IDENTIFIER ::= { private 1 }
flintstones (4242) OBJECT IDENTIFIER ::= { enterprises 4242 }
fsys (1) OBJECT IDENTIFIER ::= { flintstones 1 }
fmail (2) OBJECT IDENTIFIER ::= { flintstones 2 }
fwidgets (3) OBJECT IDENTIFIER ::= { flintstones 3 }
fprinters (4) OBJECT IDENTIFIER ::= { flintstones 4 }
fdisk (5) OBJECT IDENTIFIER ::= { flintstones 5 }
fprocess (6) OBJECT IDENTIFIER ::= { flintstones 6 }
fconfig (7) OBJECT IDENTIFIER ::= { flintstones 7 }
```

MIB ツリーの構成を次の図に示します。

図 2-8 MIB ツリーの構成



次に, 各葉ノードのオブジェクト識別子を示します。

## 2. SNMP エージェントの環境設定

葉ノード	オブジェクト識別子
fsysUsers	1.3.6.1.4.1.4242.1.1.0
fmemory	1.3.6.1.4.1.4242.1.2.0
fmailNumMsgs	1.3.6.1.4.1.4242.2.1.0
fmailListMsgs	1.3.6.1.4.1.4242.2.2.0
fwidgetsPerHour	1.3.6.1.4.1.4242.3.1.0
fprintersStatus	1.3.6.1.4.1.4242.4.1.0
fdefaultPrinter	1.3.6.1.4.1.4242.4.2.0
fUID	1.3.6.1.4.1.4242.5.1.1.1.fUID <sup>1</sup>
fUserName	1.3.6.1.4.1.4242.5.1.1.2.fUID
fUserDiskSpace	1.3.6.1.4.1.4242.5.1.1.3.fUID
fUserEmailAddress	1.3.6.1.4.1.4242.5.1.1.4.fUID
fProcessID	1.3.6.1.4.1.4242.6.1.1.1.fProcessID <sup>2</sup>
fProcessName	1.3.6.1.4.1.4242.6.1.1.2.fProcessID
fInetdConf	1.3.6.1.4.1.4242.7.1.0

注 1 fUIDの値

注 2 fProcessIDの値

### (2) システム (flintagent) へのログイン

ルートユーザーとして flintagent システムにログインします。

### (3) /etc/SnmpAgent.d/snmpd.extend ファイルの作成例

/etc/SnmpAgent.d/snmpd.extend ファイルの作成例を次に示します。

作成例の /etc/SnmpAgent.d/snmpd.extend ファイルの内容は、拡張 MIB 定義サンプルファイルの /opt/OV/prg\_samples/eagent/snmpd.extend の中に提供されています。

```
FLINTSTONES-MIB DEFINITIONS ::= BEGIN

--
-- Test MIB- used for testing snmpd.ea(extensible agent)
--
--
internet          OBJECT IDENTIFIER ::= { iso(1) org(3) dod(6)
internet(1) }
enterprises       OBJECT IDENTIFIER ::= { internet private(4) 1 }
flintstones       OBJECT IDENTIFIER ::= { enterprises 4242 }
fsys              OBJECT IDENTIFIER ::= { flintstones 1 }
fmail             OBJECT IDENTIFIER ::= { flintstones 2 }
fwidgets          OBJECT IDENTIFIER ::= { flintstones 3 }
fprinters         OBJECT IDENTIFIER ::= { flintstones 4 }
fdisk             OBJECT IDENTIFIER ::= { flintstones 5 }
fprocess          OBJECT IDENTIFIER ::= { flintstones 6 }
fconfig           OBJECT IDENTIFIER ::= { flintstones 7 }

fsysUsers OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "List of users on the flintstone machine
```

```

        READ-COMMAND: /usr/bin/users; exit 0
        READ-COMMAND-TIMEOUT: 5"
 ::= { fsys 1 }

fmemory OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Amount of memory (in megabytes) on system
    APPEND-COMMUNITY-NAME: true
    FILE-NAME: /opt/OV/prg_samples/eagent/memory"
 ::= { fsys 2 }

fmailNumMsgs OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Message count on the mail queue.
    READ-COMMAND: /usr/bin/mailq | fgrep -v Mail | wc -l"
 ::= { fmail 1 }

fmailListMsgs OBJECT-TYPE
SYNTAX DisplayString
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "List of messages on the mail queue.
    READ-COMMAND: /usr/bin/mailq
    READ-COMMAND-TIMEOUT: 10"
 ::= { fmail 2 }

fwidgetsPerHour OBJECT-TYPE
SYNTAX Gauge
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Number of widgets produced per hour.
    READ-COMMAND: /opt/OV/prg_samples/eagent/num_widgets $i $c
$o $s
    READ-COMMAND-TIMEOUT: 2
    WRITE-COMMAND: /opt/OV/prg_samples/eagent/
change_num_widgets $*
    WRITE-COMMAND-TIMEOUT: 10"
 ::= { fwidgets 1 }

fprintersStatus OBJECT-TYPE
SYNTAX INTEGER {
    up(1),
    down(2)
}
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Status of printer scheduler.
    READ-COMMAND: ps -ef|grep lpsched | grep -v grep | wc |
    awk '{ if ($1 == 0) print 2; else print 1 }'"

```

## 2. SNMP エージェントの環境設定

```
 ::= { fprinters 1 }

fdefaultPrinter OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "Default printer
         FILE-NAME: /usr/spool/lp/default"
 ::= { fprinters 2 }

fUserDiskTable OBJECT-TYPE
    SYNTAX SEQUENCE OF FUserDiskEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "List of users and the number of kilobytes in their
         home directory.
         FILE-NAME: /opt/OV/prg_samples/eagent/user_disk_space"
 ::= { fdisk 1 }

fUserDiskEntry OBJECT-TYPE
    SYNTAX FUserDiskEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "This macro documents the column that uniquely
         describes each row."
    INDEX { fUID }
 ::= { fUserDiskTable 1 }

FUserDiskEntry ::=
    SEQUENCE {
        fUID INTEGER,
        fUserName DisplayString,
        fUserDiskSpace INTEGER,
        fUserEmailAddress DisplayString
    }

fUID OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "User's unique ID"
 ::= { fUserDiskEntry 1 }

fUserName OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "User login name"
 ::= { fUserDiskEntry 2 }

fUserDiskSpace OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
```

```

STATUS mandatory
DESCRIPTION
    "Amount of disk space (in kilobytes) used by the user."
::= { fUserDiskEntry 3 }

fUserEmailAddress OBJECT-TYPE
SYNTAX DisplayString
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Email address for the user"
::= { fUserDiskEntry 4 }

fUserRootProcessTable OBJECT-TYPE
SYNTAX SEQUENCE OF FUserRootProcessEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
    "List of root processes. Do not execute command more
    than every 60 seconds.
    FILE-COMMAND: /opt/OV/prg_samples/eagent/get_processes
    FILE-COMMAND-FREQUENCY: 60
    FILE-NAME: /opt/OV/prg_samples/eagent/root_processes"
::= { fprocess 1 }

fUserRootProcessEntry OBJECT-TYPE
SYNTAX FUserRootProcessEntry
ACCESS not-accessible
STATUS mandatory
DESCRIPTION
    "This macro documents the column that uniquely
    describes each row."
INDEX { fProcessID }
::= { fUserRootProcessTable 1 }

FUserRootProcessEntry ::=
SEQUENCE {
    fProcessID INTEGER,
    fProcessName DisplayString
}

fProcessID OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Process ID"
::= { fUserRootProcessEntry 1 }

fProcessName OBJECT-TYPE
SYNTAX DisplayString
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Name of process"
::= { fUserRootProcessEntry 2 }

fInetdConf OBJECT-TYPE

```

## 2. SNMP エージェントの環境設定

```
SYNTAX DisplayString
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "The configuration file for inetd
    FILE-COMMAND: /opt/OV/prg_samples/eagent/update_inetd
$r
    FILE-COMMAND-FREQUENCY: 7200
    FILE-NAME: /etc/inetd.conf"
 ::= { fconfig 1 }

END
```

### (a) MIB オブジェクト定義例の説明

/etc/SnmpAgent.d/snmpd.extend ファイル作成例に含まれる MIB オブジェクト定義ごとに定義内容を説明します。

#### 1. システムを使用しているユーザーをリストアップする。

SNMP エージェントは、マネージャーからの GetRequest を受信した場合にコマンドを実行できます。実行するコマンドは、READ-COMMAND ラベルに記述します。/usr/bin/users コマンドを使用して、システムを使用しているユーザーのリストを MIB の値とする定義例を次に示します。

```
fsysUsers OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "List of users on the flintstone machine
        READ-COMMAND: /usr/bin/users; exit 0
        READ-COMMAND-TIMEOUT: 5"
 ::= { fsys 1 }
```

SNMP エージェントが、マネージャーからオブジェクト識別子 flintstones.fsys.fsysUsers.0 に対して GetRequest を受信すると、READ-COMMAND ラベルに記述された /usr/bin/users コマンドを実行して、その結果をマネージャーに応答します。

SNMP エージェントは、デフォルトでは READ-COMMAND ラベルに記述されたコマンドが 3 秒以内に応答しない場合、マネージャーに genErr をエラー応答します。このコマンドの終了待ち時間は、READ-COMMAND-TIMEOUT ラベルを使用して任意の時間に変更できます。コマンドの終了待ち時間を 5 秒に設定したい場合、READ-COMMAND-TIMEOUT ラベルに次のように記述します。

```
READ-COMMAND-TIMEOUT: 5
```

#### 2. 各マシンのメモリーの内容を参照する。

SNMP エージェントはほかのマシンのデバイスやアプリケーションに代わってマネー

ジャーからの要求に対してプロキシとして応答することができます。SNMP をサポートしていない 3 台のマシンのメモリーの内容を SNMP エージェントが応答するようにしたい場合について説明します。

SNMP エージェントを 3 台のマシンにプロキシとして動作させます。SNMP をサポートしていないマシンを *larry*, *curly*, *moe* と名付けます。次に示す三つのファイルの内容はそれぞれのマシンのメモリー内容を保持しています。

```
/opt/OV/prg_samples/eagent/memory.larry
/opt/OV/prg_samples/eagent/memory.curly
/opt/OV/prg_samples/eagent/memory.moe
```

このプロキシを実装する場合、次のようにオブジェクトを定義します。

```
fmemory OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Amount of memory (in megabytes) on system
        APPEND-COMMUNITY-NAME: true
        FILE-NAME: /opt/OV/prg_samples/eagent/memory"
    ::= { fsys 2 }
```

SNMP エージェントは、マネージャーからの *flintstones.fsys.fmemory* に対する SNMP Request を *larry*, *curly*, *moe* に代わって応答します。どのマシンへの SNMP Request かは、その SNMP Request に含まれているコミュニティ名から判断します。例えば、*flintstones.fsys.fmemory.0* にコミュニティ名が *moe* で GetRequest を受信した場合、SNMP エージェント機能は */opt/OV/prg\_samples/eagent/memory.moe* を読み込み、その内容をマネージャーに応答します。例からもわかるように、プロキシとして使用する場合は *APPEND-COMMUNITY-NAME:true* を記述します。SNMP エージェントは *APPEND-COMMUNITY-NAME:true* が記述されると *FILE-NAME* ラベルに記述されたファイル名にコミュニティ名を結合して（この場合 */opt/OV/prg\_samples/eagent/memory.コミュニティ名*）ファイル名とします。すでに SNMP エージェントに組み込まれているオブジェクトに対してプロキシとして動作させる場合、例えば、MIB-II オブジェクトでもプロキシとして使用できます。プロキシマシンの *Larry*, *curly*, *moe* からそれぞれの *sysDescr* を応答させたい場合について、例を用いて説明します。各マシンの *sysDescr* の内容を次のファイルに保持しています。

```
/opt/OV/prg_samples/eagent/sysDescr.larry
/opt/OV/prg_samples/eagent/sysDescr.curly
/opt/OV/prg_samples/eagent/sysDescr.moe
```

このプロキシを実装する場合、ユーザーは次のようにオブジェクトを定義します。

```
sysDescr OBJECT-TYPE
```

## 2. SNMP エージェントの環境設定

```
SYNTAX DisplayString (SIZE(0..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "A textual description of the entity. This value
    should include the full name and version
    identification of the system's hardware type,
    software operating-system, and networking
    software. It is mandatory that this only contain
    printable ASCII characters.
    APPEND-COMMUNITY-NAME : true
    FILE-NAME: /opt/OV/prg_samples/eagent/sysDescr"
 ::= { system 1 }
```

SNMP エージェントが、マネージャーからオブジェクト識別子 `system.sysDescr.0` に対して、コミュニティ名が `moe` で `GetRequest` を受信した場合、SNMP エージェントは `/opt/OV/prg_samples/eagent/sysDescr.moe` を読み込み、マネージャーにファイルの内容を応答します。また、SNMP エージェントが、マネージャーからオブジェクト識別子 `system.sysDescr.0` に対して、コミュニティ名が `public` で `GetRequest` を受信した場合、MIB-II オブジェクトの `sysDescr` の値を応答します。

## 3. メールキューを管理する。

`READ-COMMAND` ラベルを使用してメールキューを管理する定義例を次に示します。

```
fmailNumMsgs OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Message count on the mail queue.
        READ-COMMAND: /usr/bin/mailq | fgrep -v Mail
        | wc -l"
 ::= { fmail 1 }

fmailListMsgs OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "List of messages on the mail queue.
        READ-COMMAND: /usr/bin/mailq
        READ-COMMAND-TIMEOUT: 10"
 ::= { fmail 2 }
```

## 4. 無人のシステムで1時間あたりに発生するウィジェット数を管理する。

SNMP エージェントは、マネージャーからの `SetRequest` を受信した場合にコマンドを実行できます。実行するコマンドは、`WRITE-COMMAND` ラベルに記述します。`/opt/OV/prg_samples/eagent/change_num_widgets` コマンドを使用して、無人のシステムで1時間あたりに発生するウィジェット数を変更する定義例を次に示します。

```
fwidgetsPerHour OBJECT-TYPE
    SYNTAX Gauge
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "Number of widgets produced per hour.
        READ-COMMAND: /opt/OV/prg_samples/eagent/num_widgets
                    $i $c $o $s
        READ-COMMAND-TIMEOUT: 2
        WRITE-COMMAND: /opt/OV/prg_samples/eagent/change_num_
                    widgets $*
        WRITE-COMMAND-TIMEOUT: 10"
 ::= { fwidgets 1 }
```

READ-COMMAND ラベル, および WRITE-COMMAND ラベルに記述されたコマンドに, 引数を渡せます。この引数はコマンドのあとに記述します。引数の意味については, 「2.6.3(2) スクリプトまたはプログラムの記述」の引数の説明を参照してください。

#### 5. LP スケジューラを見失わないようにする。

READ-COMMAND ラベルを使用して, LP スケジューラの状態を管理する定義例を次に示します。

```
fprintersStatus OBJECT-TYPE
    SYNTAX INTEGER {
        up(1),
        down(2)
    }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Status of printer scheduler.
        READ-COMMAND: ps -ef | grep lpsched
                    | grep -v grep | wc |
                    awk '{ if ($1 == 0) print 2; else print 1 }'"
 ::= { fprinters 1 }
```

#### 6. デフォルトのプリンタを参照する。

SNMP エージェントは, ファイルの内容を MIB の値として拡張 MIB オブジェクトを定義できます。ファイルは FILE-NAME ラベルに記述します。

/usr/spool/lp/default ファイルを読み込んで, デフォルトのプリンタを参照するオブジェクトは次のように定義します。

```
fdefaultPrinter OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "Default printer
        FILE-NAME: /usr/spool/lp/default"
 ::= { fprinters 2 }
```

## 2. SNMP エージェントの環境設定

SNMP エージェントが、マネージャーからオブジェクト識別子 `flintstones.fprinters.fdefaultPrinter.0` に対して `GetRequest` を受信すると、`FILE-NAME` ラベルに記述された `/usr/spool/lp/default` ファイルを読み込んで、マネージャーに応答します。

7. ユーザー ID、ユーザーが使用しているディスクスペース、ユーザーの email のアドレスリストを参照する。

ファイルの内容を MIB の値とすることで、テーブル形式の MIB を SNMP エージェントに追加できます。ファイルの内容を MIB の値とする場合の例を示します。

ユーザー ID とそのユーザーが使用しているディスクスペース、ユーザーの email のアドレスリストが、`/opt/OV/prg_samples/eagent/user_disk_space` ファイルに格納されています。データの例を次に示します。

# User ID	User Name	Disk Space	Email Address
100	zach	120	zach@server1
201	alice	65	alice@server2
320	john	2	john@server3
119	craig	500	root@server1
217	steve	75	steve@server1
83	bob	111	bob@bobby

例で示すテーブルには、6 個の行と 4 個の列があります。テーブル形式の MIB を定義する場合、列要素の中からどの行かを特定できるような、ユニークな値を持つオブジェクトを選択する必要があります。この列要素を接尾辞と呼びます。この例では最初の列の User ID がユニークな値となっています。User Name がユニークの場合、第 2 列を接尾辞として使用します。

このテーブル形式の MIB オブジェクトは、次のように定義します。

```
fUserDiskTable OBJECT-TYPE
    SYNTAX SEQUENCE OF FUserDiskEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "List of users and the number of kilobytes in their
        home directory.
        FILE-NAME: /opt/OV/prg_samples/eagent/
        user_disk_space"
    ::= { fdisk 1 }

fUserDiskEntry OBJECT-TYPE
    SYNTAX FUserDiskEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "This macro documents the column that uniquely
        describes each row."
    INDEX { fUID }
    ::= { fUserDiskTable 1 }
FUserDiskEntry ::=
```

```

SEQUENCE {
    fUID INTEGER,
    fUserName DisplayString,
    fUserDiskSpace INTEGER,
    fUserEmailAddress DisplayString
}

fUID OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "User's unique ID"
    ::= { fUserDiskEntry 1 }

fUserName OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "User login name"
    ::= { fUserDiskEntry 2 }

fUserDiskSpace OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Amount of disk space (in kilobytes) used by the
user."
    ::= { fUserDiskEntry 3 }

fUserEmailAddress OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "Email address for the user"
    ::= { fUserDiskEntry 4 }

```

最初の OBJECT-TYPE マクロは fUserDiskTable オブジェクトと関連のあるファイルの説明をしています。また、第 2 の OBJECT-TYPE マクロは fUserDiskEntry 列中で行を特定するオブジェクトを説明しています。次のエントリーの fUserDiskEntry はテーブルの列要素を表します。最後の 4 個の OBJECT-TYPE マクロは列要素のそれぞれについて定義します。

SNMP エージェントが、マネージャーからオブジェクト識別子

fUserDiskTable.fUserDiskEntry.fUID に対して getNextRequest を受信した場合、/opt/OV/prg\_samples/eagent/user\_disk\_space ファイルを読み込み、接尾辞 (INDEX 節のオブジェクト) をキーにしてテーブルをソートします。ソートしたテーブルは次のようになります。

```

83    bob      111    bob@bobby
100   zach     120    zach@server1

```

## 2. SNMP エージェントの環境設定

```
119 craig 500 root@server1
201 alice 65 alice@server2
217 steve 75 steve@server1
320 john 2 john@server3
```

この結果から、SNMP エージェントはテーブル内の最初の値を応答します。これは第 1 行の第 1 列の値であり、マネージャーは、オブジェクト識別子

flintstones.fdisk.fUserDiskTable.fUserDiskEntry.fUID.83 の MIB の値として 83 を受け取ります。次に SNMP エージェントがオブジェクト識別子

fUserDiskTable.fUserDiskEntry.fUID.83 に対して GetNextRequest を受信した場合、SNMP エージェントはファイルが更新されていないかチェックし、ファイルが更新されている場合は、/opt/OV/prg\_samples/eagent/user\_disk\_space ファイルを再読み込みします。そして 2 行目の User ID を MIB の値として応答します。

SNMP エージェントが、オブジェクト識別子

flintstones.fdisk.fUserDiskTable.fUserDiskEntry.fUID.320 に対して

GetNextRequest を受信した場合、これ以上 user ID の行がないため第 2 列の最初の値を応答します。オブジェクト識別子は

flintstones.fdisk.fUserDiskTable.fUserDiskEntry.fUserName.83 で、その MIB の値は bob です。

SNMP エージェントが、オブジェクト識別子

flintstones.fdisk.fUserDiskTable.fUserDiskEntry.fUserEmailAddress.217 に対して GetRequest を受信した場合、Email Address の列の 217 行目を検索し、MIB の値として steve@server1 を応答します。

alice の email address を alice@server2 から alice@mailier に変更する場合、オブジェクト識別子

flintstones.fdisk.fUserDiskTable.fUserDiskEntry.fUserEmailAddress.201 と MIB の値 alice@mailier を指定して SetRequest を発行します。SNMP エージェントは、/opt/OV/prg\_samples/eagent/user\_disk\_space ファイル中のテーブルの内容を指定された値に書き換えます。

### 8. ルートプロセスの一覧を出力する。

SNMP エージェントは、snmpd.extend ファイル中の DESCRIPTION 節の

FILE-COMMAND ラベルに UNIX コマンドを記述することで、ファイルを読み込む前に UNIX コマンドを実行できます。ルートプロセスの一覧を参照する場合について、FILE-COMMAND を記述した定義例を示します。

```
fUserRootProcessTable OBJECT-TYPE
    SYNTAX SEQUENCE OF FUserRootProcessEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "List of root processes. Do not execute command more
        than every 60 seconds.
        FILE-COMMAND: /opt/OV/prg_samples/eagent
                    /get_processes
```

```

        FILE-COMMAND-FREQUENCY: 60
        FILE-NAME: /opt/OV/prg_samples/eagent/
root_processes"
    ::= { fprocess 1 }

fUserRootProcessEntry OBJECT-TYPE
    SYNTAX FUserRootProcessEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "This macro documents the column that uniquely
        describes each row."
    INDEX { fProcessID }
    ::= { fUserRootProcessTable 1 }

FUserRootProcessEntry ::=
    SEQUENCE {
        fProcessID INTEGER,
        fProcessName DisplayString
    }

fProcessID OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Process ID"
    ::= { fUserRootProcessEntry 1 }

fProcessName OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "Name of process"
    ::= { fUserRootProcessEntry 2 }

```

SNMP エージェントがマネージャーから次の GetNextRequest を受け取った場合、SNMP エージェントは、まず FILE-COMMAND に記述された /opt/OV/prg\_samples/eagent/get\_process コマンドを実行します。

```
fprocess.fuserRootProcessTable.fUserRootProcessEntry.fProcessID
fprocess.fuserRootProcessTable.fUserRootProcessEntry.fProcessName
```

このコマンドが /opt/OV/prg\_samples/eagent/root\_process ファイルを作成します (内容は root で起動されたプロセスの ID と名前です)。コマンドが終了すると SNMP エージェントは、/opt/OV/prg\_samples/eagent/root\_process ファイルを読み込み、テーブルの内容をソートしてテーブルの第 1 行、第 1 列の値をマネージャーに応答します。

SNMP エージェントが、マネージャーから GetNextRequest を受信した場合、デフォルトでは /opt/OV/prg\_samples/eagent/get\_process コマンドを最後に実行してからの時間が 10 秒以内の場合、このコマンドを実行しないで、前回読み込んだファイ

## 2. SNMP エージェントの環境設定

ルの内容をマネージャーに応答します。この周期は FILE-COMMAND-FREQUENCY ラベルを使用して任意の時間に変更できます。1 時間ごとにコマンドを実行させたい場合、FILE-COMMAND-FREQUENCY ラベルに次のように記述します。

```
FILE-COMMAND-FREQUENCY: 3600
```

snmpd.extend 中の DESCRIPTION 節の PIPE-IN-NAME ラベルと PIPE-OUT-NAME ラベルを記述することで、FILE-COMMAND ラベルに記述した UNIX コマンドの代わりに、UNIX プロセスと通信することができます。SNMP エージェントは、ファイルを読み込む前に PIPE-OUT-NAME ラベルに記述されたファイルにデータを書き込みます。UNIX プロセスはそのデータを読み込み、FILE-NAME ラベルに記述されたファイルの内容を書き換えたあと、その完了を PIPE-IN-NAME に記述されたファイルで SNMP エージェントに通知します。FILE-COMMAND ラベルの例と同様に、ルートプロセスの一覧を参照する場合について PIPE-IN-NAME ラベルと PIPE-OUT-NAME ラベルを記述した定義例を次に示します。定義例に示していない定義は、FILE-COMMAND ラベルを使用した UNIX コマンドの例と同じです。

```
fUserRootProcessTable OBJECT-TYPE
    SYNTAX SEQUENCE OF FUserRootProcessEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "List of root processes.
        PIPE-IN-NAME: /tmp/fifo_in
        PIPE-OUT-NAME: /tmp/fifo_out
        FILE-NAME: /opt/OV/prg_samples/eagent/
root_processes"
    ::= { fprocess 1 }
```

SNMP エージェントがマネージャーから次のオブジェクトの GetNextRequest を受信した場合、SNMP エージェントは、まず /tmp/fifo\_out ファイルにデータを書き込みます。

```
fprocess.fUserRootProcessTable.fUserRootProcessEntry.fProcessID
fprocess.fUserRootProcessTable.fUserRootProcessEntry.fProcessName
```

データの内容については、「2.6.2(2) テーブル形式の拡張 MIB オブジェクト」の pipe\_out\_name に渡すデータの説明を参照してください。

UNIX プロセスはこのメッセージを読み込み、/opt/OV/prg\_samples/eagent/root\_process ファイルを作成します。内容は root によって起動されたプロセスの ID とプロセスの名前です。そのあと、UNIX プロセスは /tmp/fifo\_in ファイルに 0 を書き込み、ファイルの作成が成功したことを SNMP エージェントに通知します。

SNMP エージェントは、`/tmp/fifo_in` ファイルの内容を読み込み、内容が 0 なら `/opt/OV/prg_samples/eagent/root_processes` ファイルを読み込み、内容をソートしてテーブルの第 1 行、第 1 列の値をマネージャーに応答します。

SNMP エージェントがマネージャーから `GetRequest` または `GetNextRequest` を受信した場合、デフォルトでは `PIPE-OUT-NAME` ラベルに記述されたファイルに最後にデータを書き込んでからの時間が 10 秒以内ならば新しく書き込みはしないで、前回読み込んだファイルの内容をマネージャーに応答します。この周期は `PIPE-FREQUENCY` ラベルを使用して任意の時間に変更できます。

SNMP エージェントがマネージャーから `SetRequest` を受信した場合、SNMP エージェントは `FILE-NAME` に記述されたファイルを読み込む前後で `PIPE-OUT-NAME` ラベルに記述されたファイルにデータを書き込みます。UNIX プロセスは、`PIPE-OUT-NAME` ラベルに記述されたファイルからデータを受信して必要な処理を実行したあと、`PIPE-IN-NAME` ラベルに記述されたファイルにデータ (0) を書き込みます。

#### 9. `inetd (1M)` の構成ファイルの内容を変更する。

SNMP エージェントがマネージャーから `SetRequest` を受信した場合、`FILE-COMMAND` ラベルに UNIX コマンドが記述されていると、SNMP エージェントは、`FILE-NAME` ラベルに記述されたファイルの内容を書き換える前後で UNIX コマンドを実行します。これを利用して `inetd (1M)` の構成ファイルの内容を変更する場合の定義例を次に示します。

```
fInetdConf OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "The configuration file for inetd
        FILE-COMMAND: /opt/OV/prg_samples/eagent
        /update_inetd $r
        FILE-COMMAND-FREQUENCY: 7200
        FILE-NAME: /etc/inetd.conf"
    ::= { fconfig 1 }
```

SNMP エージェントが、マネージャーからオブジェクト識別子

`flintstones.fconfig.fInetdConf.0` と MIB の値が `inetd.conf` 構成ファイルの内容である `SetRequest` を受信した場合、SNMP エージェント機能は次の順序で動作します。

- `/opt/OV/prg_samples/eagent/update_inetd SetRequest` コマンドを実行します。
- `/etc/inetd.conf` ファイルを `SetRequest` で指定された Set 値へ書き換えます。
- `/opt/OV/prg_samples/eagent/update_inetd PostSetRequest` コマンドを実行します。

ここで `/opt/OV/prg_samples/eagent/update_inetd` コマンドは最初の引数をチェックし、引数が `PostSetRequest` の場合、`/opt/OV/prg_samples/eagent/update_inetd` コマンドは `/etc/inetd -c` を実行します。このコマンドによって `/etc/inetd` は構成ファイル

## 2. SNMP エージェントの環境設定

を再度読み込みます。

## 2.7 拡張トラップの定義

この節では、snmptrap コマンドを使用して SNMP エージェントからマネージャーに SNMP トラップを送信する方法について説明します。説明する内容を次に示します。

- 拡張トラップの定義方法
- 拡張トラップの利用方法
- スクリプト例

拡張トラップを送信できるエージェントを構成するには、まず、トラップとは何かを理解する必要があります。トラップについては、「1.2.2 SNMP トラップの発行」および RFC1157:A Simple Network Management Protocol (SNMP) を参照してください。

### 2.7.1 拡張トラップの定義方法

ユーザー独自の拡張トラップを定義するには、ユーザーのトラップをユニークに識別する必要があります。このためには、一般トラップ EnterpriseSpecific (6) とユーザー固有のトラップ番号を結合します。このユーザー固有のトラップ番号の最大値は  $2^{32}-1$  です。この組み合わせで、マネージャーはトラップの種類を区別します。また、任意の情報を渡せます。

### 2.7.2 拡張トラップの利用方法

SNMP エージェントは、snmptrap コマンドを使用してトラップを送信します。また、シェルスクリプトから snmptrap コマンドを実行してトラップを送信できます。

マネージャーがエージェントの状態を監視するには、次に示す二つの方法があります。

- マネージャーから継続的にエージェントをポーリングして情報を得る方法
- エージェントからマネージャーにトラップを送信する方法

ポーリングするとネットワークには大量のトラフィックが発生します。このため、ポーリング直後に異常が発生しても、マネージャーは長時間その内容を検知できないことが考えられます。snmptrap コマンドを使用すれば、ネットワーク上の SNMP トラフィックを減少でき、より早く異常を検知できます。

なお、NNM では [ イベント設定 ] の選択および snmptrap コマンドを組み合わせることで、ユーザーの環境をカスタマイズできます。例えば、SNMP エージェントが動作するシステムの特定のプロセスに不調が発生したときに snmptrap コマンドを実行するスクリプトを SNMP エージェントに記述した場合、[ イベント設定 ] を選択して、SNMP エージェントからその特定のトラップを受け取った場合に対処できるようになります。

### 2.7.3 スクリプト例

ネットワークに接続されたプリンタのプリンタスケジューラが不調になった場合、snmptrap コマンドを実行します。snmptrap コマンドのスクリプトの例を説明します。定義例を次に示します。

```
#!/bin/sh
#
#
# This script checks to see if the printer scheduler (lpshed) is
# running. This check is performed every hour. If the scheduler
# is not
# running, the agent sends an SNMP trap to the management station.
#
# If a management station receives a trap from a system with
enterprise
# equal to .1.3.6.1.4.1.4242, generic-trap equal to 6, and the
specific trap
# equal to 2, the management station knows that the printer scheduler
for that
# agent-addr is down.
#
# The agent sends how many hours the lp scheduler has been down with
the trap.
#
AGENT_ADDRESS=15.6.71.223
MGMT_STATION=flcndmak
hours=0
while true
do
    sleep 3600
    pid='ps -ef | grep lpshed | grep -v grep |wc -l'
    if [ $pid -eq 0 ]
    then
        hours='expr $hours + 1'
        snmptrap -cpublic $MGMT_STATION .1.3.6.1.4.1.4242
$AGENT_ADDRESS 6
                2 0 ¥
                .1.3.6.1.4.1.4242.4.2.0 Integer $hours
    else
        hours=0
    fi
done
```

#### 注意事項

AIX で cron や /etc/inittab から起動されたシェルスクリプトやプログラムの延長で snmptrap または systemtrap コマンドを実行した場合、次のようなエラーが出力されて、コマンドが失敗する場合があります。

```
snmptrap:cannot set locale($LANG="Ja_JP")
```

この場合、環境変数 LC\_ALL に使用する言語を設定してください。  
使用する言語に 'C' を設定する場合の B シェルの例を次に示します。

```
LC_ALL=C  
export LC_ALL  
snmptrap f1cndmak .1.3.6.1.4.1.4242 15.6.71.223 6 2 0
```

## 2.8 クラスタ環境で運用する場合の設定

SNMP エージェントはクラスタ環境での運用に対応しています。

運用管理サーバでは、主系、従系のサーバを、個々のサーバ（ノード）として管理します。したがって、SNMP エージェントは主系と従系を常に起動させておき、フェールオーバーに対応した設定をする必要はありません。

なお、SNMP エージェントをクラスタシステムで運用するためには、次の設定が必要です。

共有ディスクの監視に必要な設定（Linux，Linux(IPF) の場合）

SNMP エージェントをクラスタシステムにインストールした場合に、監視マネージャーから共有ディスクのファイルシステム情報が取得できないときがあります。原因は、対象となるファイルシステムが `/etc/fstab` に記載されていないためです。これは、SNMP エージェントはファイルシステム情報の取得に OS のシステムコールを使用していますが、このシステムコールは `/etc/fstab` に記載されているファイルシステムだけを対象としているためです。この対策としては、`/etc/fstab` に共有ディスクのファイルシステム情報を追加します。

SNMP エージェントは、共有ディスクを監視するにあたり、`/etc/fstab` に記述された共有ディスクのフィールドのうち、第 1 フィールド（ブロックスペシャルデバイス）と第 2 フィールド（マウントポイント）のみを意識しており、その他のフィールドの設定値について、設定値が何であるかは意識していません。

そのため、`/etc/fstab` に記述された共有ディスクの他のフィールド部分に関して具体的な設定方法や設定値は、ご使用のクラスタソフトおよび OS のドキュメントを参照してください。

例えば、ご使用のクラスタ管理ソフトが HA モニタの場合で、共有ディスクのマウントポイントが `/mnt/test` であり、その共有ディスクを制御する際の要件として、OS 起動時などに共有ディスクが自動マウントされないようにする場合の、`/etc/fstab` の設定例を以下に示します。

（設定例）

```
/dev/sdb1 /mnt/test ext3 defaults,noauto 0 0
```

上記は、HA モニタを使用している場合の設定例になります。設定を行う際は、HA モニタのマニュアルおよびリリースノートをご確認頂き、設定を行ってください。

その他のクラスタソフトを使用している場合は、ご使用のクラスタソフトや OS 環境の要件に合わせて適切な設定を行ってください。

不正な共有ディスク容量の応答抑止の設定（AIX，Linux，Linux (IPF) の場合）

SNMP エージェントは共有ディスクがアンマウントされている状態で、MIB 取得要求を受けると、不正な共有ディスク容量を応答します。その抑止のため、`/etc/SnmpAgent.d/esafilesys.conf` の設定が必要です。`/etc/SnmpAgent.d/esafilesys.conf` の設定に関する詳細につきましては「4.3.2 日立企業固有 MIB オブジェクトの内容」の「(20) fileSystem64 グループ」を参照してください。

次に `/etc/SnmpAgent.d/esafilesys.conf` の記述例を示します。

(例)

記述例は、共有ディスク：`/mnt/test`，ファイル：`esatest` になります。

```
check: /mnt/test esatest
```

PowerHA(HACMP) を使用する際の設定

SNMP エージェントを PowerHA(HACMP) 環境で動作させる場合には、SNMP エージェントの SNMP リクエスト受信ポートを `161/udp` から空いている UDP ポートに変更して、AIX が提供する `snmpd` プロセスが `161/udp` を使用するように設定してください。

変更の理由としては、PowerHA(HACMP) 環境下では AIX が提供するプロセスが SNMP Request を使用して情報の交換をしているためです。

以下の例では、SNMP エージェントの SNMP 受信ポートを `8161/udp` に変更しています。

(1) NNMi または NNM の SNMP リクエストポートの変更

SNMP リクエストポートの変更については、NNMi をご使用の場合は NNMi のヘルプを、また NNM をご使用の場合は NNM のヘルプを参照して下さい。

(2) SNMP エージェントの SNMP 受信ポートの変更

以下の作業をスーパーユーザーで実行します。

1. SNMP エージェントを停止します。

`/usr/CM2/ESA/bin/snmpstop` を実行します。

2. AIX の提供する `snmpd` の SNMP 受信ポートの変更

- ・ AIX 5L V5.3/V6.1 でネイティブエージェントが SNMP v1 エージェントの場合 `/etc/services` ファイルの `snmp` 列の値が `161/udp` になっているのを確認してください。161/udp 以外の場合は変更してください。

- ・ AIX 5L V5.3/V6.1 でネイティブエージェントが SNMP v3 エージェントの場合 `/usr/CM2/ESA/opt/SnmpNative` を以下のように変更してください。

```
SNMP_NATIVE_OPTIONS="-p 161"
```

- ・ `/etc/environment` ファイルに「SNMP\_PORT=」を定義している場合は以下のように変更してください。

## 2. SNMP エージェントの環境設定

```
SNMP_PORT=161
```

3. SNMP エージェントの SNMP 受信ポートを変更します。

以下の例では SNMP エージェントの SNMP 受信ポートを 8161/udp に変更する場合で説明します。

ポート番号は 8161/udp である必要はありません。

スーパーユーザーで以下の作業を実行します。

/usr/CM2/ESA/opt/SnmpMaster に次の 2 行を追加してください。

```
SR_SNMP_TEST_PORT=8161
```

```
export SR_SNMP_TEST_PORT
```

4. SNMP エージェントのネイティブエージェントアダプターの SNMP リクエスト送信ポート変更

スーパーユーザーで /usr/CM2/ESA/opt/SnmpNaa ファイルを編集します。

以下の 2 行をファイルの最後に追加してください。

```
SNMP_NAA_OPTIONS="-port 161"
```

```
export SNMP_NAA_OPTIONS
```

上記設定を有効にするため、以下のコマンドを実行して SNMP エージェントを起動して下さい。

```
/usr/CM2/ESA/bin/snmpstart
```

## 2.9 セットアップ時の注意事項

SNMP エージェントをセットアップする場合の OS 共通の注意事項について説明します。OS 固有の注意事項については、次項以降を参照してください。

ネットワーク環境設定での一般的な注意事項

- SNMP エージェントは自ホスト名に対する IP アドレスを自ホストの IP アドレスとしますので自ホスト名の設定が必要です。
- `/etc/SnmpAgent.d/snmpd.conf` の `trap-dest:` に定義されているホスト名は IP アドレスに変換できます。

なお、DNS に対応した設定は特に必要ありません。

ファイアウォールを介してシステム間の通信をする場合の注意事項

- マネージャーシステムと SNMP エージェントの間にファイアウォールがある環境では、SNMP プロトコルを通過させるように設定してください。SNMP エージェントは、通常 161/udp ポートで SNMP リクエストを受信し、マネージャーホストの 162/udp ポートに対して SNMP トラップを送信します。

ファイアウォールに追加する設定については、「付録 B.2 ファイアウォールの通過方向」を参照してください。

SNMP エージェントの SNMP リクエストを受信するポートを変更した場合は、ファイアウォールの設定も変更してください。SNMP エージェントが自ホストで使用しているポートについては、「付録 B.1 SNMP エージェントで使用するポート番号」を参照してください。

- SNMP エージェントが SNMP トラップを送るマネージャーホストのポートは、162/udp です。SNMP エージェントを 07-50 以前のバージョンから上書きインストールするときの SNMP 受信ポートの注意事項については、「2.3 インストール時の注意事項」を参照してください。
- サーバ側監視プログラムが NAT またはファイアウォールを挟んだネットワークに存在するときは、JP1/Cm2/Internet Gateway Server を使用することを推奨します。

自ホストのホスト名変更時の注意事項

SNMP エージェントのインストール後にホスト名を変更した場合で、`sysName` の値を変更したホスト名にしたいときは、「4.1.2 標準 MIB オブジェクトの内容」の「(1) System グループ」を参照してください。

マネージャーシステムの IP アドレスまたはホスト名変更時の注意事項

構成定義ファイル (`/etc/SnmpAgent.d/snmpd.conf`) の `trap-dest:` に定義されている IP アドレスまたはホスト名を見直してください。

`/etc/hosts` ファイルに関する注意事項

HP-UX (IPF) 以外の OS で `naaagt` プロセスを使用する場合は、`/etc/hosts` ファイルに、次の例に示すように `localhost` に対する IP アドレスを設定してください。

## 2. SNMP エージェントの環境設定

(例)

127.0.0.1 localhost

naaagt プロセスは起動時に自ホストのネイティブエージェントと通信するための IP アドレスを OS の関数を使用して「localhost」をキーに検索します。そこで、IP アドレスの検索に失敗するとネイティブエージェントと通信できないため、naaagt プロセスは終了します。

ファイルシステムに関する注意事項

ファイルシステム情報としては次に示す MIB があります。

- FileSystem グループ (general 2)  
詳細については、「4.2.2 HP 企業固有 MIB オブジェクトの内容」の「(2) FileSystem グループ」を参照してください。
- fileSystem64 グループ (hiux 21)  
詳細については、「4.3.2 日立企業固有 MIB オブジェクトの内容」の「(20) fileSystem64 グループ」を参照してください。

これらの MIB で、ある特定のファイルシステム情報を応答しないようにする設定ができます。詳細については、「4.3.2 日立企業固有 MIB オブジェクトの内容」の「(20) fileSystem64 グループ」を参照してください。

### 2.9.1 セットアップ時の注意事項 (Linux および Linux (IPF) の場合)

SNMP エージェントをセットアップする場合の、Linux および Linux (IPF) 固有の注意事項について説明します。OS 共通の注意事項については、「2.9 セットアップ時の注意事項」を参照してください。

SNMP 受信ポート番号のデフォルト値に関する注意事項

SNMP エージェントは SNMP 受信ポート番号のデフォルト値を 161/udp ではなく 22161/udp にしています。これに合わせて、マネージャーのリモートポートの設定を変更してください。マネージャーの SNMP の設定については、マネージャーのドキュメントを参照してください。

SNMP 受信ポート番号のデフォルト値の変更理由

Linux のネイティブエージェントの SNMP 受信ポート番号が 161/udp から別の番号に変更するには OS 全体のセキュリティレベルを下げる必要があります。しかし、OS 全体のセキュリティレベルは下げられないため Linux のネイティブエージェントの SNMP 受信ポート番号は 161/udp のままで使用します。そこで SNMP エージェントが SNMP 受信ポート番号を 161/udp を使用するとポート番号の二重使用になるため、ポート番号が取得できません。したがって SNMP エージェントの SNMP 受信ポート番号を 22161/udp にしています。

# 3

## SNMP エージェントの運用

この章では、SNMP エージェントの起動と停止および SNMP エージェントの起動中の運用について説明します。

---

3.1 SNMP エージェントの起動と停止

---

3.2 SNMP エージェントの SNMP 受信ポートの変更

---

3.3 最大接続サブエージェント数の変更

---

3.4 バックアップとリストア

---

3.5 運用上の注意事項

---

## 3.1 SNMP エージェントの起動と停止

---

SNMP エージェントの起動と停止の方法について説明します。

### 3.1.1 SNMP エージェントの起動

SNMP エージェントのマスターエージェントやサブエージェントは、通常、システムの起動とともに自動的に起動されます。システムの起動時に実行されるファイルについては、「(5) システムの起動時に実行されるファイル」を参照してください。

手動で起動する場合は、スーパーユーザーで `snmpstart` コマンドを実行します。`snmpstart` コマンドの詳細については、「5. コマンドとプロセス」の「`snmpstart`」を参照してください。

#### 注意事項

システムの OS が Linux, Linux (IPF) の場合は、ネイティブエージェントを起動してから `snmpstart` コマンドを実行してください。

SNMP エージェントの各プロセスはインストール時のデフォルトの状態で作動できますが、SNMP エージェントが提供するプロセスの起動オプションおよび環境変数の定義をご使用の環境に合わせて設定することもできます。

次に、SNMP エージェントが提供するプロセスの起動オプションおよび環境変数の定義を常に有効にするための設定（カスタマイズ）について説明します。

#### (1) プロセスの起動オプションのカスタマイズおよび環境変数の定義

SNMP エージェントを構成するエージェントは、そのエージェントのプロセスの起動オプションおよび環境変数を定義するファイル（環境変数定義ファイル）を持っています。プロセスの起動オプションおよび環境変数の定義は環境変数定義ファイルに設定します。なお、プロセスの起動オプションは環境変数で指定します。そのため、特に断りがない場合、環境変数にプロセスの起動オプションを含みます。

環境変数定義ファイルに設定した環境変数の値は、SNMP エージェント起動時に有効になります。

次に、環境変数定義ファイルの設定手順を示します。

#### 1. SNMP エージェントを停止する。

SNMP エージェントの停止については、「3.1.2 SNMP エージェントの停止」を参照してください。

#### 2. 環境変数を定義する。

環境変数を定義します。

SNMP エージェントが提供する環境変数定義ファイルについては、「(2) SNMP エージェントが提供する環境変数定義ファイル」を参照してください。

プロセスの起動オプションを指定する環境変数については、「(3) 環境変数定義ファイルで指定できる起動オプション」を、プロセスの起動オプション以外の環境変数については、「(4) プロセスで指定できる環境変数」を参照してください。

### 3. SNMP エージェントを起動する。

システムの起動または `snmpstart` コマンドを実行すると、SNMP エージェントが起動されます。

SNMP エージェントの各プロセスの環境変数の詳細については、「5. コマンドとプロセス」の「プロセス」を参照してください。

#### 参考

引数を指定して SNMP エージェントのプロセスを起動した場合、そのプロセスが停止するまで指定が有効です。プロセスのオプションを起動時に常に有効にするためには、環境変数で起動オプションを指定してください。

## (2) SNMP エージェントが提供する環境変数定義ファイル

SNMP エージェントはプロセスの環境変数定義ファイルを提供しています。プロセスの環境変数定義ファイル、およびそのファイルのディレクトリ名を次の表に示します。

表 3-1 SNMP エージェントが提供するプロセスの環境変数定義ファイル

SNMP エージェントが提供するプロセス	ファイル名	ディレクトリ名
<code>snmpdm</code>	<code>SnmpMaster</code>	Solaris の場合 /etc/rc.config.d AIX の場合 /usr/CM2/ESA/opt HP-UX (IPF), Linux, Linux (IPF) の場合 /opt/CM2/ESA/opt
<code>extsubagt</code>	<code>SnmpExtAgt</code>	
<code>hp_unixagt</code>	<code>SnmpHpunix</code>	
<code>htc_unixagt1</code>	<code>SnmpHtcunix1</code>	
<code>htc_unixagt2</code>	<code>SnmpHtcunix2</code>	
<code>htc_unixagt3</code>	<code>SnmpHtcunix3</code>	
<code>htc_monagt1</code>	<code>SnmpHtcMonagt1</code>	
<code>naaagt</code>	<code>SnmpNaa</code>	
<code>trapdestagt</code>	<code>SnmpTrpDst</code>	

## (3) 環境変数定義ファイルで指定できる起動オプション

プロセスの起動オプションは起動オプション環境変数で指定してください。プロセスの起動オプション環境変数とその環境変数で指定できるオプションを、次の表に示します。

### 3. SNMP エージェントの運用

表 3-2 プロセスの起動オプションを指定する環境変数と指定できるオプション

SNMP エージェントが提供するプロセス	起動オプション環境変数	指定できるオプション
snmpdm	SNMP_MASTER_OPTIONS	-aperror, -apwarm, -apverbose, -authfail, -Contact, -hexdump, -Location, -mask, -sysDescr, -tcplocal, -vbdump
extsubagt	SNMP_EXTAGT_OPTIONS	-e, -E, -aperror, -apwarn, -apconfig, -appacket, -aptrap, -apaccess, -apemanate, -apverbose, -apuser, -retry, -fcmdguard, -pipeguard, -invokeid
hp_unixagt	SNMP_HPUNIX_OPTIONS	-aperror, -apwarn, -apconfig, -appacket, -aptrap, -apaccess, -apemanate, -apverbose, -apuser, -retry
htc_unixagt1	SNMP_HTCUNIX1_OPTIONS	-aperror, -apwarn, -apconfig, -appacket, -aptrap, -apaccess, -apemanate, -apverbose, -apuser, -retry
htc_unixagt2	SNMP_HTCUNIX2_OPTIONS	-aperror, -apwarn, -apconfig, -appacket, -aptrap, -apaccess, -apemanate, -apverbose, -apuser, -retry
htc_unixagt3	SNMP_HTCUNIX3_OPTIONS	-aperror, -apwarn, -apconfig, -appacket, -aptrap, -apaccess, -apemanate, -apverbose, -apuser, -retry
htc_monagt1	SNMP_HTCMONAGT1_OPTIONS	-i, -t, -s, -d
naaagt	SNMP_NAA_OPTIONS	-aperror, -apwarn, -port, -readcomm, -writecomm, -timeout
trapdestagt	SNMP_TRAPDEST_OPTIONS	-aperror, -apwarn, -apconfig, -appacket, -aptrap, -apaccess, -apemanate, -apverbose, -apuser, -retry

注 プロセスの起動オプションの詳細については、「5. コマンドとプロセス」の「プロセス」を参照してください。なお、snmpdm プロセスの起動オプション環境変数 SNMP\_MASTER\_OPTIONS で指定できる -aperror オプションおよび -apwarm オプションは、それぞれ snmpdm プロセスの -mask オプションのログマスク値 (FACTORY\_WARN, FACTORY\_ERROR) に対応します。

extsubagt プロセスの起動オプションを常に有効にするための設定例を次に示します。

(例)

システム起動時または snmpstart コマンド実行時に、extsubagt プロセスの起動オプション -fcmdguard, -pipeguard, -invokeid が有効になるように設定します。適用 OS が Solaris の場合、環境変数定義ファイル (/etc/rc.config.d/SnmpExtAgt) の SNMP\_EXTAGT\_OPTION 環境変数に、次のオプションを設定します。

- FILE\_COMMAND で指定したコマンドの応答監視時間を 15 秒

- PIPE\_IN\_NAME, PIPE\_OUT\_NAME で指定したパイプの応答監視時間を 25 秒
- パイプによるデータの送受信データの一致判定に識別番号を使用する

```
SNMP_EXTAGT_OPTIONS="-fcmddguard 15 -pipeguard 25 -invokeid"
export SNMP_EXTAGT_OPTIONS
```

これ以降、システムの起動時または /opt/CM2/ESA/bin/snmpstart を実行すると、このオプションが有効となります。

#### (4) プロセスで指定できる環境変数

SNMP エージェントが提供するプロセスの環境変数定義ファイルには、次の表に示す環境変数が指定できます。なお、プロセスの起動オプション環境変数については、表 3-2 を参照してください。

表 3-3 環境変数定義ファイルに指定できる環境変数

SNMP エージェントが提供するプロセス	指定できる環境変数
snmpdm	SR_SNMP_TEST_PORT SNMP_HTC_AUTH_LOG SR_TRAP_TEST_PORT SNMP_HTC_INIT_WAIT_TIME SNMP_HTC_SNMPD_LOG_SIZE SNMP_HTC_SNMPD_LOG_CNT
extsubagt	SR_SNMP_TEST_PORT
hp_unixagt	SR_SNMP_TEST_PORT SNMP_HTC_SOLARIS_SWAP_RESERVED ( Solaris の場合 ) SNMP_HTC_AIX_EXCEPT_FILECACHE ( AIX の場合 )
htc_unixagt1	SR_SNMP_TEST_PORT SNMP_HTC_FILE_EXTEND
htc_unixagt2	SR_SNMP_TEST_PORT
htc_unixagt3	SR_SNMP_TEST_PORT
htc_monagt1	SNMP_HTCMONAGT1_START SNMP_HTC_AIX_CPU_SMT ( AIX の場合 )
naaagt	SR_SNMP_TEST_PORT
trapdestagt	SR_SNMP_TEST_PORT

注 環境変数の詳細については、「5. コマンドとプロセス」の「プロセス」の「外部への影響」の説明を参照してください。

#### (5) システムの起動時に実行されるファイル

システムの起動時に実行されるファイルを、OS 別に次に示します。

Solaris の場合

### 3. SNMP エージェントの運用

/etc/rc2.d/S97esa から /opt/CM2/ESA/bin/snmpstart コマンドが実行されます。

AIX の場合

/etc/inittab に登録されている起動シェルスクリプト /usr/CM2/ESA/bin/esa が実行されます。

Linux , Linux (IPF) の場合

システムのランレベルに応じて、次のファイルのどれかから /opt/CM2/ESA/bin/snmpstart コマンドが実行されます。

- /etc/rc.d/rc2.d/S55esa
- /etc/rc.d/rc3.d/S55esa
- /etc/rc.d/rc5.d/S55esa

HP-UX (IPF) の場合

/sbin/rc2.d/S560esa から /opt/CM2/ESA/bin/snmpstart コマンドが実行されます。

#### 3.1.2 SNMP エージェントの停止

マスターエージェントやサブエージェントは、通常、システムの停止とともに自動的に停止されます。システム停止時に実行されるファイルについては、「(2) システムの停止時に実行されるファイル」を参照してください。

手で停止する場合は、スーパーユーザーで snmpstop コマンドを実行します。

snmpstop コマンドの詳細については、「5. コマンドとプロセス」の「snmpstop」を参照してください。

なお、SNMP エージェントのプロセスを個別に手で停止する方法を次に示します。

1. ps コマンドを使用して、マスターエージェントやサブエージェントのプロセス番号を調査する。
2. スーパーユーザーで、kill コマンドに「1. で調査したプロセス番号」を指定して実行する。
3. ps コマンドを使用して、マスターエージェントやサブエージェントが停止しているかどうかを確認する。

##### (1) プロセスを個別に停止する場合の注意事項

SNMP エージェントのプロセスを個別に停止する場合の注意事項を次に示します。

- マスターエージェントを停止した場合、マスターエージェントに接続しているサブエージェントが終了することがあります。また、OS やほかのプログラムが提供しているサブエージェントも停止することもあります。
- マスターエージェントを再起動した場合、サブエージェントも必ず起動するか、またはシステムを再起動してください。

## (2) システムの停止時に実行されるファイル

システムの停止時に実行されるファイルを次に示します。

Solaris の場合

次のファイルから /opt/CM2/ESA/bin/snmpstop コマンドが実行されます。

- /etc/rc0.d/K02esa
- /etc/rc1.d/K02esa

AIX の場合

/etc/rc.shutdown から /usr/CM2/ESA/bin/snmpstop コマンドが実行されます。

Linux , Linux (IPF) の場合

次のファイルから /opt/CM2/ESA/bin/snmpstop コマンドが実行されます。

- /etc/rc.d/rc0.d/K65esa
- /etc/rc.d/rc2.d/K65esa
- /etc/rc.d/rc3.d/K65esa
- /etc/rc.d/rc5.d/K65esa
- /etc/rc.d/rc6.d/K65esa

HP-UX (IPF) の場合

/sbin/rc1.d/K440esa から /opt/CM2/ESA/bin/snmpstop コマンドが実行されます。

### 3.1.3 ネイティブエージェントの起動と停止

SNMP エージェントが提供する snmpstart コマンドおよび snmpstop コマンドでのネイティブエージェントの起動と停止は OS によって異なります。

Solaris および AIX の場合

snmpstart コマンドでネイティブエージェントも起動されます。また、snmpstop コマンドでネイティブエージェントも停止されます。

Linux および Linux (IPF) の場合

snmpstart コマンドおよび snmpstop コマンドでは、ネイティブエージェントの起動・停止は実行されません。ネイティブエージェントの起動および停止は OS 側の制御に依存します。

## 3.2 SNMP エージェントの SNMP 受信ポートの変更

次に示す場合、SNMP エージェントの SNMP 受信ポートを変更する必要があります。

- ネイティブエージェントや他社製品の SNMP エージェントと、SNMP エージェントが SNMP 受信ポートを同時に使用する。
- ファイアウォール環境で SNMP エージェントの SNMP 受信ポートを変更する。

なお、SNMP エージェントの SNMP 受信ポートは次の順番で決まります。

1. 環境変数 `SR_SNMP_TEST_PORT` に指定されているポート番号  
OS が Solaris または AIX の場合、`SR_SNMP_TEST_PORT` の値は 161 が、Linux、Linux IPF では 22161 がそれぞれデフォルトで指定されています。そのほかの OS の場合は指定されていません。
2. `/etc/services` ファイルの `snmp` サービス名に指定されているポート番号

SNMP エージェントおよびネイティブエージェントの SNMP 受信ポートの変更方法について、次に示します。

### 3.2.1 SNMP エージェントの SNMP 受信ポートの変更方法

ここでは、SNMP エージェントの SNMP 受信ポートの変更方法について説明します。

SNMP エージェントの SNMP 受信ポートの変更方法には、`SR_SNMP_TEST_PORT` による方法と `/etc/services` ファイルによる方法の二つがありますが、`/etc/services` ファイルの値を変更すると、ほかの SNMP エージェントに影響が生じるおそれがあります。そのため変更方法としては、`SR_SNMP_TEST_PORT` を使用する方法をお勧めします。

次に、`SR_SNMP_TEST_PORT` を使用した変更方法について説明します。

1. 次の 2 行を環境変数を定義するファイルに追加する。

```
SR_SNMP_TEST_PORT=変更後のポート番号
export SR_SNMP_TEST_PORT
```

`SR_SNMP_TEST_PORT` 環境変数を定義するファイルはシステムの OS によって異なります。定義するファイルを次に示します。

Solaris の場合：`/etc/rc.config.d/SnmpMaster`

AIX の場合：`/usr/CM2/ESA/opt/SnmpMaster`

HP-UX (IPF)、Linux、Linux (IPF) の場合：`/opt/CM2/ESA/opt/SnmpMaster`

2. UDP ポート番号を変更する (Solaris、AIX、Linux、Linux (IPF) の場合)。

ネイティブエージェントの SNMP 受信ポートを 8161/udp または 161/udp からほかのポート番号に変更する場合、naaagt プロセスがネイティブエージェントに接続するための UDP ポート番号についても変更が必要です。

次の 2 行を環境変数を定義するファイルに追加する。

```
SNMP_NAA_OPTIONS="-port ネイティブエージェントのSNMP受信ポート番号"
export SNMP_NAA_OPTIONS
```

SNMP\_NAA\_OPTIONS 環境変数を定義するファイルはシステムの OS によって異なります。定義するファイルを次に示します。

Solaris の場合：etc/rc.config.d/SnmpNaa

AIX の場合：/usr/CM2/ESA/opt/SnmpNaa

Linux, Linux (IPF) の場合：/opt/CM2/ESA/opt/SnmpNaa

### 3. SNMP エージェントを再起動する。

スーパーユーザーで次のコマンドを実行します。

```
/opt/CM2/ESA/bin/snmpstart
```

## 3.2.2 ネイティブエージェント snmpd の SNMP 受信ポートの変更 (AIX の場合)

ここでは、AIX が提供するネイティブエージェント snmpd の SNMP 受信ポートの変更方法について説明します。

SNMP エージェントをインストールした場合、ネイティブエージェント snmpd の SNMP 受信ポートは 8161/udp になります。8161/udp から別のポート番号に変更したい場合は、次の方法で変更してください。なお、作業は SNMP エージェントを停止させてから、スーパーユーザーで実行してください。

#### 1. 環境変数 SNMP\_NATIVE\_OPTIONS を変更する。

SnmpNative 環境変数定義ファイルを vi などのエディタで開き、次に示す行の 8161 を任意のポート番号に変更する。

```
SNMP_NATIVE_OPTIONS="-p 8161"
```

#### 2. snmpstart コマンドを実行する。

SNMP エージェントが再起動されます。SNMP エージェントを再起動することでネイティブエージェント snmpd も再起動されます。

なお、ネイティブエージェント snmpd が SNMP v1 エージェントの場合は、ネイティブエージェント snmpd が、SNMP 受信ポートとして 161/udp を使用して、SNMP エー

### 3. SNMP エージェントの運用

エージェントの SNMP 受信ポートは 161/udp 以外のポートを使用する必要があります。

環境変数 `SNMP_NATIVE_OPTIONS` は、ネイティブエージェント `snmpd` が SNMP v3 エージェントの場合に使用します。

## 3.3 最大接続サブエージェント数の変更

---

最大接続サブエージェント数とは、マスターエージェントが接続できるサブエージェントの数です。デフォルトでは、この値は 22 です。この値は、`/etc/srconf/agt/snmpd.cnf` ファイルに定義されています。マスターエージェントは、接続できるサブエージェント数以上の接続要求を受信すると、`/var/adm/snmpd.log` ファイルに次のメッセージを出力します。

```
AllocSubagent: runtime_MAX_SUBAGENTS exceeded
```

このメッセージが出力された場合、最大接続サブエージェント数を変更します。最大接続サブエージェント数を設定するには、次に示す手順をスーパーユーザーで実行してください。

1. `/etc/srconf/agt/snmpd.cnf` を開く。
2. 次の行を検索する。

```
MAX_SUBAGENTS 22
```

この値は最大接続サブエージェント数です。
3. `MAX_SUBAGENTS` ラベルの後ろに、接続するサブエージェント数を記述する。
4. 次の行を検索する。

```
MAX_THREADS 22
```

この値は、マスターエージェントが同時に生成するスレッドの最大数です。
5. `MAX_THREADS` ラベルの後ろに、`MAX_SUBAGENTS` ラベルに指定した値と同じ値を指定する。
6. `snmpstart` コマンドを実行する。  
SNMP エージェントが再起動されます。

### 注意事項

`MAX_SUBAGENTS`、`MAX_THREADS` の最小値は 22 です。最大値は OS が一つのプロセスで生成できる最大スレッド数に依存します。

## 3.4 バックアップとリストア

---

この節ではバックアップとリストアについて説明します。

バックアップおよびリストアのコマンドは提供していませんので、任意の方法で実施してください。バックアップソフトウェアを使用してフルバックアップ・フルリストアを実施する場合の注意事項については、「3.4.2 フルバックアップ・フルリストア時の注意事項」を参照してください。

### 3.4.1 設定ファイルのバックアップとリストア

障害回復などの目的でバックアップデータを取得する場合の設定ファイルを次に示します。

- /etc/SnmpAgent.d/snmpd.conf
- /etc/srconf/agt/snmpd.cnf
- /etc/srconf/agt/naa.cnf ( HP-UX (IPF) の場合は必要ありません )
- /etc/SnmpAgent.d/esafilesys.conf
- /etc/SnmpAgent.d/esalocale.conf

また、必要に応じて次のファイルのバックアップデータを取得してください。

ユーザー定義 MIB 機能を使用している場合

/etc/SnmpAgent.d/snmpd.extend

SNMP エージェントの起動オプションを変更した場合

/etc/rc.config.d/Snmp で始まるファイル ( Solaris の場合 )

/opt/CM2/ESA/opt ディレクトリ配下のファイル ( HP-UX (IPF) , Linux , Linux (IPF) の場合 )

/usr/CM2/ESA/opt ディレクトリ配下のファイル ( AIX の場合 )

/opt/CM2/ESA/ext ディレクトリ配下のファイル ( AIX 以外の場合 )

/usr/CM2/ESA/ext ディレクトリ配下のファイル ( AIX の場合 )

バックアップは SNMP エージェントが停止中でも動作中でも実施できます。リストアは必ず SNMP エージェントを停止させてから実施してください。

### 3.4.2 フルバックアップ・フルリストア時の注意事項

バックアップソフトウェアを使用してフルバックアップ・フルリストアを実施する場合の注意事項を次に示します。注意事項を事前に十分に検証して、フルバックアップ・フルリストアを実施してください。

- SNMP エージェントを停止した状態で、フルバックアップ・フルリストアを実行してください。

SNMP エージェントが動作中の状態でフルバックアップを実行すると、動作中に生成するファイルのバックアップに失敗することがあります。

- フルバックアップを実行すると、SNMP エージェントが動作中に取得した情報も含めてバックアップされます。

フルバックアップ時に SNMP エージェントが取得できていた情報と、フルリストア後に SNMP エージェントが取得できる情報が異なる場合、不整合が生じ、フルリストア後の SNMP エージェントが正しく動作できないことがあります。

なお、フルバックアップを実行してマシンとは別のマシンへフルリストアを実行した場合、不整合が生じる可能性が高くなります。

- フルリストア後に SNMP エージェントが正しく動作しない場合は、SNMP エージェントを停止して、次の操作を実行してください。
  1. SNMP エージェントの設定ファイルをバックアップする。
  2. SNMP エージェントをアンインストールする。
  3. SNMP エージェントを新規にインストールする。
  4. バックアップしていた設定ファイルをリストアする。

## 3.5 運用上の注意事項

SNMP エージェントの OS 共通の運用上の注意事項について説明します。OS 固有の注意事項については、次項以降を参照してください。

### 単調増加ファイルの注意事項

動作開始以降に単調増加するログファイルの注意事項を次の表に示します。

表 3-4 単調増加するログファイルの注意事項

パス	ログを出力するプロセス	注意事項
/tmp/esa.log	インストール/アンインストール時のシェル	この製品のインストール/アンインストール時でなければ削除しても問題ありません。

なお、単調増加ファイルの容量の上限はありません。

### ログファイルの注意事項

動作開始以降に出力するログファイルの注意事項を次の表に示します。

表 3-5 ログファイルの注意事項

パス <sup>1</sup>	ログを出力するプロセス	注意事項
/var/adm/snmpd.logn	snmpd	デフォルトでは、一つのログファイルサイズを 10MB として、10 面 (n: 1 ~ 10) でラップアラウンドしてログファイルを取得します。運用上、デフォルト値が問題となる場合は、ログファイルサイズ、面数、および出力先パスを変更してください <sup>2</sup> 。

注 1 AIX の場合は /usr/adm 配下です。

注 2 ログファイルサイズ、面数、出力先パスの変更については「6.3 ログの取得」を参照してください。

### SNMP エージェントが使用しているファイルについての注意事項

/tmp/.AgentSockets ディレクトリおよびその配下のファイルは、SNMP エージェントが使用しているファイルです。SNMP エージェントが起動中は削除しないでください。

SNMP エージェントが停止中の場合は削除しても問題ありません。SNMP エージェントは起動時にこのディレクトリを作成します。

### ノードの IP アドレスを変更した場合の注意事項

SNMP エージェントの起動中に、そのノードの IP アドレスを変更した場合は、SNMP エージェントを再起動してください。

### 環境変数定義ファイルのバックアップについての注意事項

- 環境変数定義ファイルをバックアップする場合、Snmp で始まるファイル名でバックアップファイルを作成しないでください。バックアップファイルに付ける名称の例を次に示します。

(例) /opt/CM2/ESA/opt/SnmpMaster のバックアップファイルの場合

/opt/CM2/ESA/opt/Bak.SnmpMaster

- システムの OS が Solaris の場合、/etc/rc.config.d 配下に環境変数定義のバックアップファイルを作成しないでください。

#### ネイティブエージェントアダプター機能についての注意事項

Solaris, AIX, Linux, Linux(IPF) のシステムでは、SNMP エージェントのネイティブエージェントアダプターと OS が提供するネイティブエージェント間のコミュニティ名を一致させてください。

#### SNMP エージェントの実行権限についての注意事項

SNMP エージェントのファイルのアクセス権は root だけで実行できます。ファイルのアクセス権は変更しないでください。

#### SNMP エージェントの言語環境

SNMP エージェントはどの言語環境でも、出力するメッセージは英語だけです。また、SNMP エージェントをインストールしたあとにシステムの言語環境を変更しても問題ありません。

#### JP1/CM2/SSO を使用してリソースを収集する場合の注意事項

- プラットフォームごとにタイムアウト値およびリトライ回数を指定してください。OS ごとに推奨するタイムアウト値を次の表に示します。なお、タイムアウト値は、システム負荷やネットワーク環境によって異なります。使用する環境に合ったタイムアウト値を設定してください。

表 3-6 推奨するタイムアウト値

OS	推奨するタイムアウト値
Solaris	6.0 秒以上
AIX	3.0 秒以上
Linux, Linux (IPF)	3.0 秒以上
HP-UX (IPF)	0.8 秒以上

SNMP リクエストは UDP を使用していますが、UDP にはリトライ機能がないため、リトライ回数を必ず指定してください。

- NNM で企業固有 MIB を取得する場合は、MIB の注意事項を確認してください。MIB の注意事項については、「4.2.2 HP 企業固有 MIB オブジェクトの内容」および「4.3.2 日立企業固有 MIB オブジェクトの内容」のグループの注意事項を参照してください。

### 3. SNMP エージェントの運用

#### システム時刻を変更する場合の注意事項

システム時刻を進める場合、特に作業は必要ありません。  
システム時刻を戻す場合は、次の手順を実行してください。

1. `snmpstop` コマンドで SNMP エージェントを停止する。
2. システム時刻を変更する。
3. `snmpstart` コマンドで SNMP エージェントを再起動する。

#### OS 起動時の `coldStart` トラップ送信についての注意事項

マスターエージェントの `snmpdm` プロセスは、デフォルトでは起動されてから 15 秒後に `coldStart` トラップを送信します。

ほかのサブエージェントの起動完了を確認することなく `coldStart` トラップは送信されるため、この間マネージャーからの要求に対しては応答しません。通常、この 15 秒間はサブエージェントが起動完了するのに十分な時間ですが、ご使用の環境によっては起動完了が間に合わないことがあります。その場合は、`SnmppMaster` ファイルの `SNMP_HTC_INIT_WAIT_TIME` 環境変数に `coldStart` トラップを送信するまでの時間（単位は秒）を指定して、`coldStart` トラップ送信のタイミングを調整してください。

`SNMP_HTC_INIT_WAIT_TIME` 環境変数の指定例を次に示します。

(例)

```
SNMP_HTC_INIT_WAIT_TIME=15
export SNMP_HTC_INIT_WAIT_TIME
```

SIGHUP 受信時の構成定義ファイル (`/etc/SnmppAgent.d/snmpd.conf`) の再読み込みについての注意事項

SNMP エージェントの動作中に、SIGHUP 受信時の構成定義ファイル (`/etc/SnmppAgent.d/snmpd.conf`) の再読み込みは実施しません。

#### JP1/Cm2/SSO から論理 IP アドレスを監視する際の注意事項

- AIX の場合  
クラスタソフトとして日立の HA モニタを使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。
- Linux の場合  
クラスタソフトとして日立の HA モニタ、CLUSTERPRO X、Veritas Cluster Server を使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。
- Linux (IPF) の場合  
クラスタソフトとして日立の HA モニタを使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。
- HP-UX (IPF) の場合  
クラスタソフトとして日立の HA モニタおよび HP Serviceguard を使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。

### 3.5.1 運用上の注意事項 (Solaris の場合)

システムの OS が Solaris である SNMP エージェントの、Solaris 固有の運用上の注意事項について説明します。OS 共通の注意事項については、「3.5 運用上の注意事項」を参照してください。

#### SMF 機能を適用した Solaris10 での運用上の注意事項

snmpstop コマンドを実行すると、SNMP エージェントの停止に伴って、SMF 機能によって管理されるネイティブエージェントのプロセス (snmpd および snmpdx) も停止します。snmpstart コマンドを実行すると、SNMP エージェントの起動に伴って、SMF 機能によって管理されるネイティブエージェントのプロセス (snmpd および snmpdx) も起動します。このためネイティブエージェントのプロセス (snmpd および snmpdx) を手動で起動 / 停止する必要はありません。

#### スワップ空間サイズの取得方法についての注意事項

SNMP エージェントが取得する Solaris のデバイススワップ空間のサイズは、デフォルトでは予約済みの値は含まれていません。デバイススワップ空間のサイズを予約済みの値を含む値で取得する場合、Snmphpunix ファイルの

SNMP\_HTC\_SOLARIS\_SWAP\_RESERVED 環境変数に Y を指定します。

なお、予約済みの値とは現在は割り当てられていないが、あとから使用できるようにメモリーによって回収されるスワップ空間のサイズのことです。

SNMP\_HTC\_SOLARIS\_SWAP\_RESERVED 環境変数の指定例を次に示します。

(例)

```
SNMP_HTC_SOLARIS_SWAP_RESERVED=Y
export SNMP_HTC_SOLARIS_SWAP_RESERVED
```

### 3.5.2 運用上の注意事項 (AIX の場合)

システムの OS が AIX である SNMP エージェントの、AIX 固有の運用上の注意事項について説明します。OS 共通の注意事項については、「3.5 運用上の注意事項」を参照してください。

#### AIX の MIB page グループの取得時の注意事項

システム構成によっては、日立企業固有 MIB の page グループの MIB を取得しようとする、エラーが頻発する場合があります。この場合、まず、SNMP エージェントが提供する page.exe コマンドの実行時間を計測してください。

page.exe コマンドの実行が 10 秒以上掛かる場合、page グループの MIB を正しく取得することはできません。その状態のまま運用を続けると、page 情報取得時に実行される OS コマンドのプロセスが不当に動作し続け、システムに負荷を掛けるおそれがあります。

この問題を回避するために page.exe を次のように修正してください。修正をすると page グループの MIB を取得した場合も、コマンドは実行されなくなり、page グループ

### 3. SNMP エージェントの運用

ブの MIB 値としては常にページ数 0 を意味する擬似情報が返されます。

(修正前)

```
export PATH=$PATH:/usr/sbin
export LANG=C
lsps -a
```

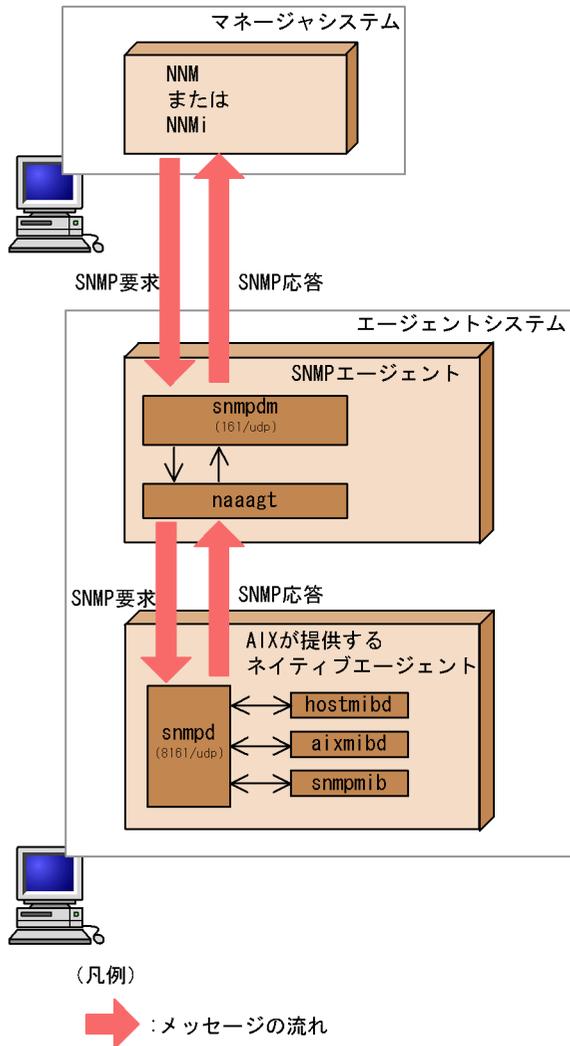
(修正後)

```
export PATH=$PATH:/usr/sbin
export LANG=C
echo
```

ネイティブエージェント機能の注意事項

- SNMP エージェントが提供するネイティブエージェントアダプター (naaagt プロセス) は、SNMPv1 の SNMP get/get-next-request にコミュニティ名「public」を指定して、AIX が提供するネイティブエージェント (snmpd プロセス) から MIB を取得します。snmpd プロセスのデフォルトの設定では、SNMPv1 でコミュニティ名「public」の SNMP get/get-next-request を許可しているので問題はありません。ただし、snmpd プロセスのコミュニティ名の設定を変更する場合は、必ず SNMPv1 の SNMP get/get-next-request を許可してください。また snmpd プロセスのコミュニティ名を「public」以外に変更した場合は、naaagt プロセスの設定も変更してください。naaagt プロセスの設定変更方法については、「2.5.3 ネイティブエージェントアダプターの設定方法」を参照してください。
- snmpd プロセスは AIX が提供する snmpv3\_ssw コマンドによって snmpdv1 エージェントを使用するか snmpdv3 エージェントを使用するの切り替えることができます。AIX のデフォルトは snmpdv3 エージェントを使用します。SNMP エージェントと snmpdv1 エージェントまたは snmpdv3 エージェントを同時に起動すると、SNMP 受信ポート (161/udp) の競合が発生します。そのため、snmpdv1 エージェントまたは snmpdv3 エージェントの SNMP 受信ポートを変更する必要があります。SNMP エージェントは、AIX のデフォルトで使用される snmpdv3 エージェントを前提としてポートを変更します。変更方法は AIX のドキュメントを参照してください。
- AIX の提供する hostmibd、aixmibd、snmpmibd の各プロセスが 161/udp を取得している snmpdm プロセスに対して、get/get-next-request を発行して MIB 値の取得をします。そのときに使用するコミュニティ名は「public」です。AIX が提供するネイティブエージェント機能の流れを次の図に示します。

図 3-1 AIX が提供するネイティブエージェント機能の流れ



SNMP エージェントをインストールしたあとに、SNMP エージェントの SNMP 受信ポート番号を変更していない場合で、コミュニティ名「public」での get/get-next-request を許可していないときは、snmpd プロセス (hostmibd, aixmibd, snmpmibd) が SNMP get/get-next-request を発行するときに使用するコミュニティ名を変更してください。変更手順を次に示します。なお、変更作業はスーパーユーザーで実行してください。

1. /usr/CM2/ESA/opt/SnmpNative を vi などのエディタで開く。
2. 次に示す行の「public」を SNMP エージェントが get/get-next-request を許可しているコミュニティ名に変更する。

### 3. SNMP エージェントの運用

```
SNMP_SNMPMIBD_OPTIONS="-c public"  
SNMP_HOSTMIBD_OPTIONS="-c public"  
SNMP_AIXMIBD_OPTIONS="-c public"
```

#### 3. SNMP エージェントを再起動する。

SNMP エージェントを再起動することで `hostmibd` , `aixmibd` , `snmpmibd` も再起動されます。次のコマンドを実行してください。

```
/usr/CM2/ESA/bin/snmpstart
```

#### OS のメモリー不足によるプロセス終了の回避についての注意事項

AIX では OS でメモリー不足が生じると SIGKILL が発行され、プロセスが終了することがあります。この現象を回避するには、SNMP エージェントを起動するユーザーの環境変数に `PSALLOC=early` を設定し、設定後、SNMP エージェントを起動してください。なお、環境変数 `PSALLOC` に `early` を設定する場合には、同時に環境変数 `NODISCLAIM=true` を設定してください。指定後に SNMP エージェントを再起動してください。

環境変数 `PSALLOC` および `NODISCLAIM` は `SnmMaster` ファイルに設定してください。

次に例を示します。

#### (例)

```
SNMP_MASTER_OPTIONS="-tcplocal"           # Master Agent options  
export SNMP_MASTER_OPTIONS  
PSALLOC=early  
export PSALLOC  
NODISCLAIM=true  
export NODISCLAIM
```

#### CPU 利用率情報の取得についての注意事項

SNMP エージェントが取得する CPU 利用率は、デフォルトでは個々の CPU の利用率を加算し CPU 数で割り算した結果の小数点以下を切り捨てた値です。

SMT 環境では、環境変数定義ファイル `SnmHtcMonagt1` で

`SNMP_HTC_AIX_CPU_SMT` 環境変数に `Y` を指定すると、マシン全体の CPU 利用率を取得できます。環境変数定義ファイルのパスは、「付録 A SNMP エージェントのファイルの一覧」を参照してください。

`SNMP_HTC_AIX_CPU_SMT` 環境変数の指定例を次に示します。

#### (例)

```
SNMP_HTC_AIX_CPU_SMT=Y  
export SNMP_HTC_AIX_CPU_SMT
```

#### 使用中の物理メモリー容量の取得についての注意事項

AIX では使用中の物理メモリーをファイルキャッシュとして利用することで、ファイ

ルアクセスを向上させています。そのため、SNMP エージェントが取得する使用中の物理メモリー容量には、デフォルトでファイルキャッシュ値が含まれています。

環境変数定義ファイル `SnmHpunix` ファイルで

`SNMP_HTC_AIX_EXCEPT_FILECACHE` 環境変数に `Y` を指定すると、ファイルキャッシュ値を含まない使用中の物理メモリー容量を取得できます。環境変数定義ファイルのパスは、「付録 A SNMP エージェントのファイルの一覧」を参照してください。

`SNMP_HTC_AIX_EXCEPT_FILECACHE` 環境変数の指定例を次に示します。

(例)

```
SNMP_HTC_AIX_EXCEPT_FILECACHE=Y
export SNMP_HTC_AIX_EXCEPT_FILECACHE
```



# 4

## MIB オブジェクト一覧

この章では、SNMP エージェントが実装している MIB オブジェクトの一覧と実装状況について説明します。

---

4.1 標準 MIB オブジェクト一覧

---

4.2 HP 企業固有 MIB オブジェクト一覧

---

4.3 日立企業固有 MIB オブジェクト一覧

---

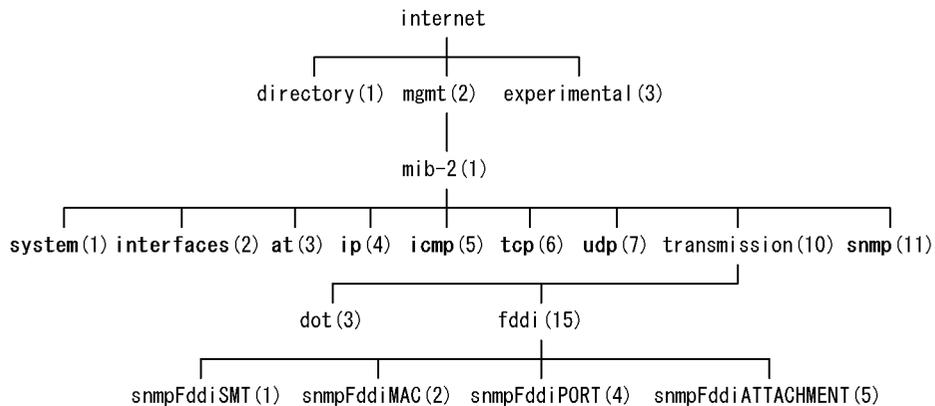
## 4.1 標準 MIB オブジェクト一覧

この節では、SNMP エージェントが実装している標準 MIB オブジェクトの一覧と実装状況について説明します。

### 4.1.1 標準 MIB オブジェクトの構成

標準 MIB オブジェクトの構成を、次の図に示します。

図 4-1 標準 MIB オブジェクトの構成



(凡例) 太字 : この節では、太字の標準 MIB オブジェクトを説明しています。

標準 MIB オブジェクトの内容および実装状況の参照先一覧を次の表に示します。

表 4-1 標準 MIB オブジェクトの参照先一覧

標準 MIB オブジェクトのグループ		参照先	
		MIB オブジェクトの内容	MIB オブジェクトの実装状況
internet.mgmt.mib-2	system	表 4-2	表 4-10
	interfaces	表 4-3	-
	at	表 4-4	-
	ip	表 4-5	-
	icmp	表 4-6	-
	tcp	表 4-7	-
	udp	表 4-8	-
	snmp	表 4-9	-

(凡例) - : 該当なし。

## 4.1.2 標準 MIB オブジェクトの内容

グループごとの標準 MIB オブジェクトの内容を表 4-2 ~ 表 4-9 に示します。なお，表中では次の凡例を使用しています。

(凡例)

- : 該当なし。

オブジェクトの内容は，RFC1213，RFC1285，または RFC1398，ならびに /var/opt/OV/share/snmp\_mibs/eagent/ 配下の rfc1213-MIB-II，rfc1285-FDDI，または rfc1398-ETHER でも参照できます。

### (1) System グループ

System グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-2 System グループ ( internet.mgmt.mib-2.system ) ( 1.2.1.1 )

ID	オブジェクト名	内容	単位
1	sysDescr	システムに関する記述。	-
2	sysObjectID	システムに与えられたオブジェクト識別子の値。	-
3	sysUpTime	システムが起動してからの経過時間。	1/100 秒
4	sysContact	システム管理者の連絡先。	-
5	sysName	システムのホスト名。	-
6	sysLocation	システムの設置場所。	-
7	sysServices	システムが提供しているサービス ( OSI 参照レイヤ )	-

#### 注

SNMP エージェントは SNMP エージェントのインストール時のホスト名を sysName の値とします。

また，SNMP エージェントはマネージャーからの SetRequest によって設定された値を sysName の値とし，sysName の値は /etc/srconf/agt/snmpd.cnf 内に保存します。

SNMP エージェントのインストール後にホスト名が変更された場合でも，SNMP エージェントは sysName の値を新しいホスト名に変更しません。それは，マネージャーからの SetRequest によって設定された値を sysName の値としている場合があるためです。そこで，新しいホスト名を sysName の値とする方法を次に示します。どれかの方法で設定してください。

- /etc/srconf/agt/snmpd.cnf の sysName の値を新しいホスト名に設定し，/opt/CM2/ESA/bin/snmpstart コマンドを使用して，SNMP エージェントを再起動し

#### 4. MIB オブジェクト一覧

ます。

- /etc/srconf/agt/snmpd.cnf 内の「sysName ホスト名」となっている行を削除します。その後、SNMP エージェントを再起動します。
- SetRequest を使用して、sysName に新しいホスト名を設定します。

### (2) Interfaces グループ

Interfaces グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-3 Interfaces グループ (internet.mgmt.mib-2.interfaces) (1.2.1.2)

ID	オブジェクト名	内容	単位
1	ifNumber	インタフェースの数。	-
2	ifTable	インタフェースエンティティ情報テーブル。	-
2.1	ifEntry	各エントリは ifIndex の値で識別される。	-
2.1.1	ifIndex	インタフェース番号。	-
2.1.2	ifDescr	インタフェース名。	-
2.1.3	ifType	インタフェースのタイプ。 other (1), regular1822 (2), hdh1822 (3), ddn-x25 (4), rfc877x25 (5), ethernet-csmacd (6), iso88023-csmacd (7), iso88024-tokenBus (8), iso88025-tokenRing (9), iso88026-man (10), starLan (11), proteon-10Mbit (12), proteon-80Mbit (13), hyperchannel (14), fddi (15), lapb (16), sdlc (17), dsl (18), el (19), basicISDN (20), primaryISDN (21), propPintToPointSerial (22), ppp (23), softwareLoopback (24), eon (25), ethernet-3Mbit (26), nsip (27), slip (28), ultra (29), ds3 (30), sip (31), frame-relay (32)	-
2.1.4	ifMtu	インタフェースが送受信できる IP データグラムの最大長。	オクテット
2.1.5	ifSpeed	転送スピード。	ビット/秒
2.1.6	ifPhysAddress	直接 IP の下にあるプロトコルレイヤのインタフェースアドレス。	-
2.1.7	ifAdminStatus	インタフェースの望ましい状態。 up (1), down (2), testing (3)	-
2.1.8	ifOperStatus	インタフェースの現在のオペレーション状態。 up (1), down (2), testing (3)	-
2.1.9	ifLastChange	インタフェースの状態が変化した時間 (エージェントの sysUpTime の値)	1/100 秒
2.1.10	ifInOctets	インタフェースで受信した総オクテット数。	オクテット
2.1.11	ifInUcastPkts	上位プロトコルに配信したサブネットユニキャストパケット数。	パケット

ID	オブジェクト名	内容	単位
2.1.12	ifInNUcastPkts	上位プロトコルに配信したサブネットブロードキャストまたはマルチキャストのパケット数。	パケット
2.1.13	ifInDiscards	資源の制限などのために廃棄される受信パケット数。	パケット
2.1.14	ifInErrors	エラーのため上位プロトコルに配信できなかった受信パケット数。	パケット
2.1.15	ifInUnknownProtos	不明または未定義のプロトコルのため廃棄された受信パケット数。	パケット
2.1.16	ifOutOctets	インタフェース上から送出した総オクテット数。	オクテット
2.1.17	ifOutUcastPkts	上位プロトコルが要求したサブネットユニキャストアドレスへの送信パケット数。	パケット
2.1.18	ifOutNUcastPkts	上位プロトコルが要求したサブネットブロードキャストまたはマルチキャストへの送信パケット数。	パケット
2.1.19	ifOutDiscards	資源の制限などのために廃棄された出力パケット数。	パケット
2.1.20	ifOutErrors	エラーのため送信できなかった受信パケット数。	パケット
2.1.21	ifOutQLen	出力パケットキューの長さ (パケット数)	パケット
2.1.22	ifSpecific	MIB 固有のポインタ。	-

### (3) AddressTranslation グループ

AddressTranslation グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-4 AddressTranslation グループ (internet.mgmt.mib-2.at) (1.2.1.3)

ID	オブジェクト名	内容	単位
1	atTable	ネットワークアドレスの物理アドレスへの変換テーブル。	-
1.1	atEntry	各エントリは atIfIndex と atNetAddress の値で識別される。	-
1.1.1	atIfIndex	対応するインタフェース番号の ifIndex の値。	-
1.1.2	atPhysAddress	対応するインタフェースの物理アドレス。	-
1.1.3	atNetAddress	対応するインタフェースの物理アドレスに対応するネットワークアドレス。	-

### (4) IP グループ

IP グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-5 IP グループ (internet.mgmt.mib-2.ip) (1.2.1.4)

ID	オブジェクト名	内容	単位
1	ipForwarding	フォワーディングするゲートウェイとして動作するかどうか。 forwarding (1), not-forwarding (2)	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2	ipDefaultTTL	IP パケットのデフォルトの Time-To-Live 値。	-
3	ipInReceives	インタフェースから受信したデータグラム数。	データグラム
4	ipInHdrErrors	IP ヘッダエラーのために廃棄された入力データグラム数。	データグラム
5	ipInAddrErrors	誤配送のために廃棄されたデータグラム数。	データグラム
6	ipForwDatagrams	IP の最終送信先がこのエンティティではない入力データグラム数。	データグラム
7	ipInUnknownProtos	不明または未定義のプロトコルのために廃棄されたデータグラム数。	データグラム
8	ipInDiscards	資源の制限などのために廃棄された入力データグラム数。	データグラム
9	ipInDelivers	IP ユーザープロトコルに正常に配布された入力データグラム数。	データグラム
10	ipOutRequests	IP ユーザープロトコルから供給された IP データグラム数。	データグラム
11	ipOutDiscards	資源の制限のために廃棄された出力 IP データグラム数。	データグラム
12	ipOutNoRoutes	送信先への転送ルートが見つからないために破棄された IP データグラム数。	データグラム
13	ipReasmTimeout	再アセンブル待ち受信フラグメントがホールドされている最大秒数。	秒
14	ipReasmReqds	再アセンブルの必要がある IP フラグメント数。	-
15	ipReasmOKs	正常に再アセンブルされた IP フラグメント数。	-
16	ipReasmFails	再アセンブルで検出された障害の数。	-
17	ipFragOKs	正常にフラグメント化された IP データグラム数。	データグラム
18	ipFragFails	フラグメント化できなかったために廃棄された IP データグラム数。	データグラム
19	ipFragCreates	フラグメント化によって生成された IP データグラムフラグメントの数。	-
20	ipAddrTable	IP アドレスに関連するアドレッシング情報テーブル。	-
20.1	ipAddrEntry	各エントリは ipAdEntAddr の値で識別される。	-
20.1.1	ipAdEntAddr	アドレッシング情報 IP アドレス。	-
20.1.2	ipAdEntIfIndex	対応するインタフェース番号の ifIndex 値。	-
20.1.3	ipAdEntNetMask	IP アドレスに関連するサブネットマスク。	-
20.1.4	ipAdEntBeastAddr	IP アドレスに関連するインタフェースで送信するために使用される IP ブロードキャストアドレス内最下位ビット値。	-

ID	オブジェクト名	内容	単位
20.1.5	ipAdEntReasmMaxSize	受信した IP データグラムフラグメントを再アセンブルできる IP データグラムの最大長。	データグラム
21	ipRouteTable	IP アドレスに関連する IP ルーティングテーブル。	-
21.1	ipRouteEntry	各エントリは ipRouteDest の値で識別される。	-
21.1.1	ipRouteDest	ディスティネーション IP アドレス。	-
21.1.2	ipRouteIfIndex	対応するインタフェース番号の ifIndex 値。	-
21.1.3	ipRouteMetric1	プライマリルーティングメトリック。	-
21.1.4	ipRouteMetric2	代替ルーティングメトリック。	-
21.1.5	ipRouteMetric3	代替ルーティングメトリック。	-
21.1.6	ipRouteMetric4	代替ルーティングメトリック。	-
21.1.7	ipRouteNextHop	次のホップの IP アドレス。	-
21.1.8	ipRouteType	ルートのタイプ。 other ( 1 ), invalid ( 2 ), direct ( 3 ), indirect ( 4 )	-
21.1.9	ipRouteProto	ルートのルーティングメカニズム。 other ( 1 ), local ( 2 ), netmgmt ( 3 ), icmp ( 4 ), egp ( 5 ), ggp ( 6 ), hello ( 7 ), rip ( 8 ), is-is ( 9 ), es-is ( 10 ), cisco-Igrp ( 11 ), bbnSpfIgp ( 12 ), ospf ( 13 ), bgp ( 14 )	-
21.1.10	ipRouteAge	ルートが最後に更新されてからの経過時間。	秒
21.1.11	ipRouteMask	ディスティネーションアドレスと論理積を取るマスク値。	-
21.1.12	ipRouteMetric5	代替ルーティングメトリック。	-
21.1.13	ipRouteInfo	特定のルーティングプロトコルの MIB をポイントするためのオブジェクト ID。	-
22	ipNetToMediaTable	IP アドレス変換テーブル。	-
22.1	ipNetToMediaEntry	各エントリは ipNetToMediaIfIndex と ipNetToMediaNetAddress の値で識別される。	-
22.1.1	ipNetToMediaIfIndex	対応するインタフェーステーブルの ifIndex の値。	-
22.1.2	ipNetToMediaPhys Address	対応するインタフェースの物理アドレス。	-
22.1.3	ipNetToMediaNet Address	対応する IP アドレス。	-
22.1.4	ipNetToMediaType	変換の種類や状態。 other ( 1 ), invalid ( 2 ), dynamic ( 3 ), static ( 4 )	-
23	ipRoutingDiscards	破棄されたルーティングエントリ数。	エントリ

#### 4. MIB オブジェクト一覧

### (5) ICMP グループ

ICMP グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-6 ICMP グループ (internet.mgmt.mib-2.icmp) (1.2.1.5)

ID	オブジェクト名	内容	単位
1	icmpInMsgs	受信した ICMP メッセージ総数。	メッセージ
2	icmpInErrors	受信した ICMP エラーメッセージ総数。	メッセージ
3	icmpInDestUnreachs	受信した ICMP 送信先到達不可メッセージ数。	メッセージ
4	icmpInTimeExcds	受信した ICMP 時間超過メッセージ数。	メッセージ
5	icmpInParmProbs	受信した ICMP パラメタ障害メッセージ数。	メッセージ
6	icmpInSrcQuenchs	受信した ICMP Source Quench メッセージ数。	メッセージ
7	icmpInRedirects	受信した ICMP リダイレクトメッセージ数。	メッセージ
8	icmpInEchos	受信した ICMP エコ-要求メッセージ数。	メッセージ
9	icmpInEchoReps	受信した ICMP エコ-応答メッセージ数。	メッセージ
10	icmpInTimestamps	受信した ICMP タイムスタンプ要求メッセージ数。	メッセージ
11	icmpInTimestampReps	受信した ICMP タイムスタンプ応答メッセージ数。	メッセージ
12	icmpInAddrMasks	受信した ICMP アドレスマスク要求メッセージ数。	メッセージ
13	icmpInAddrMaskReps	受信した ICMP アドレスマスク応答メッセージ数。	メッセージ
14	icmpOutMsgs	送信した ICMP メッセージ総数。	メッセージ
15	icmpOutErrors	送信した ICMP エラーメッセージ数。	メッセージ
16	icmpOutDestUnreachs	送信した ICMP 送信先到達不可メッセージ数。	メッセージ
17	icmpOutTimeExcds	送信した ICMP 時間超過メッセージ数。	メッセージ
18	icmpOutParmProbs	送信した ICMP パラメタ障害メッセージ数。	メッセージ
19	icmpOutSrcQuenchs	送信した ICMP Source Quench メッセージ数。	メッセージ
20	icmpOutRedirects	送信した ICMP リダイレクトメッセージ数。	メッセージ
21	icmpOutEchos	送信した ICMP エコ-要求メッセージ数。	メッセージ
22	icmpOutEchoReps	送信した ICMP エコ-応答メッセージ数。	メッセージ
23	icmpOutTimestamps	送信した ICMP タイムスタンプ要求メッセージ数。	メッセージ
24	icmpOutTimestampReps	送信した ICMP タイムスタンプ応答メッセージ数。	メッセージ
25	icmpOutAddrMasks	送信した ICMP アドレスマスク要求メッセージ数。	メッセージ
26	icmpOutAddrMaskReps	送信した ICMP アドレスマスク応答メッセージ数。	メッセージ

## (6) TCP グループ

TCP グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-7 TCP グループ ( internet.mgmt.mib-2.tcp ) ( 1.2.1.6 )

ID	オブジェクト名	内容	単位
1	tcpRtoAlgorithm	再送タイムアウト値を決定するために使用されるアルゴリズム。 other ( 1 ), constant ( 2 ), rsre ( 3 ), vanj ( 4 )	-
2	tcpRtoMin	最小の再送タイムアウト時間。	ミリ秒
3	tcpRtoMax	最大の再送タイムアウト時間。	ミリ秒
4	tcpMaxConn	一度に開設できる最大の TCP コネクション数。	-
5	tcpActiveOpens	TCP コネクションが CLOSE から SYN-SENT に状態遷移した回数。	-
6	tcpPassiveOpens	TCP コネクションが LISTEN から SYN-SENT に状態遷移した回数。	-
7	tcpAttemptFails	SYS-SENT または SYN-RCVD から CLOSE に状態遷移した回数に, SYN-RCVD から LISTEN に状態遷移した回数を加えたもの。	-
8	tcpEstabResets	ESTABLISHED または CLOSE-WAIT から CLOSE に状態遷移した回数。	-
9	tcpCurrEstab	現在 ESTABLISHED または CLOSE-WAIT 状態の TCP コネクション数。	-
10	tcpInSegs	受信セグメント総数。	セグメント
11	tcpOutSegs	送信セグメント総数。	セグメント
12	tcpRetransSegs	再送セグメント総数。	セグメント
13	tcpConnTable	TCP コネクション情報テーブル。	-
13.1	tcpConnEntry	各エントリは, tcpConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, および tcpConnRemPort の値で識別される。	-
13.1.1	tcpConnState	TCP コネクション状態。 closed ( 1 ), listen ( 2 ), synSent ( 3 ), synReceived ( 4 ), established ( 5 ), finWait1 ( 6 ), finWait2 ( 7 ), closeWait ( 8 ), lastAck ( 9 ), closing ( 10 ), timeWait ( 11 ), deleteTCB ( 12 )	-
13.1.2	tcpConnLocalAddress	TCP コネクションのローカル側 IP アドレス。	-
13.1.3	tcpConnLocalPort	TCP コネクションのローカル側ポート番号。	-
13.1.4	tcpConnRemAddress	TCP コネクションのリモート側 IP アドレス番号。	-
13.1.5	tcpConnRemPort	TCP コネクションのリモート側ポート番号。	-
14	tcpInErrs	エラーとして受信した TCP セグメント総数。	セグメント

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
15	tcpOutRsts	RST (リセット) フラグをオンにして送信した TCP セグメント数。	セグメント

#### (7) UDP グループ

UDP グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-8 UDP グループ (internet.mgmt.mib-2.udp) (1.2.1.7)

ID	オブジェクト名	内容	単位
1	udpInDatagrams	UDP ユーザーへ配信された UDP データグラム数。	データグラム
2	udpNoPorts	送信先ポートにアプリケーションがない UDP データグラム数。	データグラム
3	udpInErrors	送信先ポートにアプリケーションがない以外の理由で配信されなかった UDP データグラム数。	データグラム
4	udpOutDatagrams	配送された UDP データグラム数。	データグラム
5	udpTable	UDP リスナー情報テーブル。	-
5.1	udpEntry	各エントリは udpLocalAddress と udpLocalPort の値で識別される。	-
5.1.1	udpLocalAddress	UDP リスナーのローカル IP アドレス。	-
5.1.2	udpLocalPort	UDP リスナーのローカルポート番号。	-

#### (8) SNMP グループ

SNMP グループの標準 MIB オブジェクトの内容を次の表に示します。

表 4-9 SNMP グループ (internet.mgmt.mib-2.snmp) (1.2.1.11)

ID	オブジェクト名	内容	単位
1	snmpInPkts	TP (トランスポートサービス) から受信した SNMP メッセージ総数。	メッセージ
2	snmpOutPkts	TP へ送信した SNMP メッセージ総数。	メッセージ
3	snmpInBadVersions	未定義のプロトコルバージョンを検出した受信メッセージ数。	メッセージ
4	snmpInBadCommunityNames	未知のコミュニティを検出した受信 SNMP メッセージ数。	メッセージ
5	snmpInBadCommunityUses	操作権限のないコミュニティを検出した受信 SNMP メッセージ数。	メッセージ
6	snmpInASNParseErrs	ASN.1 文法解析誤りを起こした受信 SNMP メッセージ数。	メッセージ

ID	オブジェクト名	内容	単位
8	snmpInTooBigs	error-status が 'tooBig' の受信 SNMP メッセージ数。	メッセージ
9	snmpInNoSuchNames	error-status が 'noSuchName' の受信 SNMP メッセージ数。	メッセージ
10	snmpInBadValues	error-status が 'BadValueno' の受信 SNMP メッセージ数。	メッセージ
11	snmpInReadOnlys	error-status が 'ReadOnly' の受信 SNMP メッセージ数。	メッセージ
12	snmpInGenErrs	error-status が 'GenErr' の受信 SNMP メッセージ数。	メッセージ
13	snmpInTotalReqVars	すべての受信 GetRequest-PDU および GetNextRequest-PDU 中の MIB 変数の数。	-
14	snmpInTotalSetVars	すべての受信 SetRequest-PDU 中の MIB 変数の数。	-
15	snmpInGetRequests	受信した GetRequest-PDU の数。	-
16	snmpInGetNexts	受信した GetNextRequest-PDU の数。	-
17	snmpInSetRequests	受信した SetRequest-PDU の数。	-
18	snmpInGetResponses	受信した GetResponse-PDU の数。	-
19	snmpInTraps	受信した Trap-PDU の数。	-
20	snmpOutTooBigs	error-status が 'tooBig' の送信 SNMP メッセージ数。	メッセージ
21	snmpOutNoSuchNames	error-status が 'noSuchName' の送信 SNMP メッセージ数。	-
22	snmpOutBadValues	error-status が 'BadValueno' の送信 SNMP メッセージ数。	-
24	snmpOutGenErrs	error-status が 'GenErr' の送信メッセージ数。	-
25	snmpOutGetRequests	送信した GetRequest-PDU の数。	-
26	snmpOutGetNexts	送信した GetNextRequest-PDU の数。	-
27	snmpOutSetRequests	送信した SetRequest-PDU の数。	-
28	snmpOutGetResponses	送信した GetResponse-PDU の数。	-
29	snmpOutTraps	送信した Trap-PDU の数。	-
30	snmpEnableAuthenTraps	エージェントが認証失敗トラップを送信できるかどうか。 enabled (1), disabled (2)	-

### 4.1.3 標準 MIB オブジェクトの実装状況

標準 MIB オブジェクトの system の実装状況を、表 4-10 に示します。

#### 4. MIB オブジェクト一覧

ネイティブエージェントの標準 MIB オブジェクト ( interfaces , at , ip , icmp , tcp , udp ) の実装状況については、各 OS のドキュメントを参照してください。なお、ネイティブエージェントの標準 MIB オブジェクトの取得方法を次に示します。

- Solaris , AIX , Linux , Linux (IPF) の場合  
 ネイティブエージェントからそれらの値を取得します。ネイティブエージェントが起動していない場合、それらの値を取得できません。
- HP-UX (IPF) の場合  
 OS 提供の mib2agt プロセスからそれらの値を取得します。

#### ( 1 ) System グループ

System グループの標準 MIB オブジェクトの実装状況を次の表に示します。

表 4-10 標準 MIB オブジェクトの実装状況 ( System グループ )  
 ( internet.mgmt.mib-2.system ) ( 1.2.1.1 )

オブジェクト名	MIB オペレーション							
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux , Linux (IPF) の場合	
	get	set	get	set	get	set	get	set
sysDescr		-		-		-		-
sysObjectID		-		-		-		-
sysUpTime		-		-		-		-
sysContact								
sysName								
sysLocation								
sysServices		-		-		-		-

( 凡例 )

- : get オペレーションで値を取得できる。または set オペレーションで値をセットできる。
- : アクセス権限がない。noSuchName で応答する。

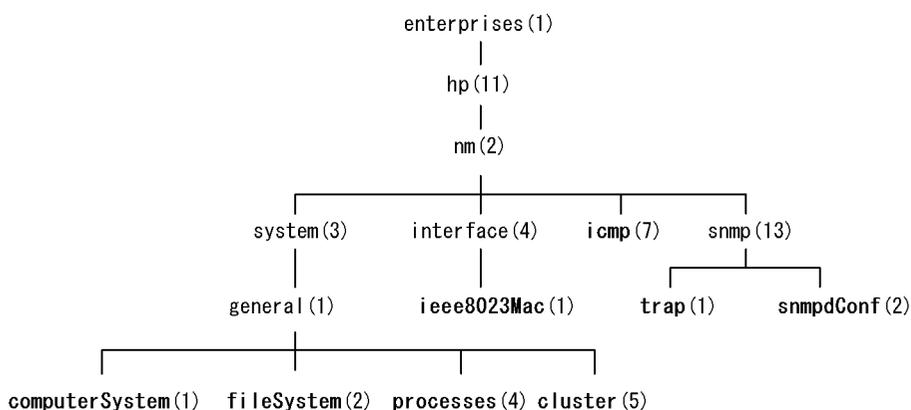
## 4.2 HP 企業固有 MIB オブジェクト一覧

この節では、SNMP エージェントが実装している HP 企業固有 MIB オブジェクトの一覧と実装状況について説明します。

### 4.2.1 HP 企業固有 MIB オブジェクトの構成

HP 企業固有 MIB オブジェクトの構成を、次の図に示します。

図 4-2 HP 企業固有 MIB オブジェクトの構成



(凡例) 太字：この節では、太字のHP企業固有MIBオブジェクトを説明しています。

HP 企業固有 MIB オブジェクトの内容および実装状況の参照先一覧を次の表に示します。

表 4-11 HP 企業固有 MIB オブジェクトの参照先一覧

HP 企業固有 MIB オブジェクトのグループ				参照先	
				MIB オブジェクトの内容	MIB オブジェクトの実装状況
enterprises.hp.nm	system	general	computerSystem	表 4-12	表 4-20
			fileSystem	表 4-13	表 4-21
			processes	表 4-14	表 4-22
			cluster	表 4-15	表 4-23
	interface	ieee8023Mac	表 4-16	表 4-24	
	icmp		表 4-17	表 4-25	
	snmp	trap	表 4-18	表 4-26	

#### 4. MIB オブジェクト一覧

HP 企業固有 MIB オブジェクトのグループ			参照先	
			MIB オブジェクトの内容	MIB オブジェクトの実装状況
		snmpdConf	表 4-19	表 4-27

### 4.2.2 HP 企業固有 MIB オブジェクトの内容

グループごとの HP 企業固有 MIB オブジェクトの内容を、表 4-12 ~ 表 4-19 に示します。なお、表中では次の凡例を使用しています。

(凡例)

- : 該当なし。

MIB オブジェクトの内容は、`/var/opt/OV/share/snmp_mibs/eagent/hp-unix` でも参照できます。

#### (1) ComputerSystem グループ

ComputerSystem グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-12 ComputerSystem グループ

(enterprises.hp.nm.system.general.computerSystem)(1.11.2.3.1.1)

ID	オブジェクト名	内容	単位
1	computerSystemUpTime	システムが起動してからの経過時間。	1/100 秒
2	computerSystemUsers	システムにログインしているユーザー数。	-
3	computerSystemAvgJobs1	過去 1 分の期間での実行待ち行列の平均ジョブ数 * 100。 実行待ち行列の平均ジョブ数とは、過去の 1 分間に「実行状態」または「実行可能状態」であったプロセスやスレッドの平均値のことです。 例えば、この実行待ち行列の平均ジョブ数が「1」であった場合、過去 1 分間で「実行状態」または「実行可能状態」となっていたプロセスやスレッドが平均的に一つ存在していて、常に CPU は処理を実行していた状況であったと推測できます。	-
4	computerSystemAvgJobs5	過去 5 分の期間での実行待ち行列の平均ジョブ数 * 100。	-
5	computerSystemAvgJobs15	過去 15 分の期間での実行待ち行列の平均ジョブ数 * 100。	-
6	computerSystemMaxProc	システムで許される最大プロセス数。	-

ID	オブジェクト名	内容	単位
7	computerSystemFreeMemory <sub>1</sub>	物理メモリーの空き容量。	キロバイト
8	computerSystemPhysMemory	物理メモリー容量。	キロバイト
9	computerSystemMaxUserMem	最大ユーザーメモリー容量。	キロバイト
10	computerSystemSwapConfig <sub>2, 3</sub>	デバイススワップ空間のサイズ。	キロバイト <sub>4</sub>
11	computerSystemEnabledSwap	使用できるディスクスワップ空間のサイズ。	キロバイト
12	computerSystemFreeSwap <sub>2</sub>	実際の空きスワップ領域のサイズ。	キロバイト <sub>4</sub>
13	computerSystemUserCPU <sub>5</sub>	nice 値 21 以上で、ユーザーモードで実行された CPU 時間。ただし、Solaris, AIX では SNMP エージェントが起動してからのユーザーモードで実行された CPU 時間。	1/100 秒
14	computerSystemSysCPU <sub>5</sub>	カーネルモードで実行された CPU 時間。ただし、Solaris, AIX では SNMP エージェントが起動してからのカーネルモードで実行された CPU 時間。	1/100 秒
15	computerSystemIdleCPU <sub>5</sub>	アイドル状態の CPU 時間。ただし、Solaris, AIX では SNMP エージェントが起動してからのアイドル状態の CPU 時間。	1/100 秒
16	computerSystemNiceCPU <sub>5</sub>	nice 値 20 以下で、ユーザーモードで実行された CPU 時間。ただし、Solaris, AIX では SNMP エージェントが起動してからのユーザーモードで実行された CPU 時間。	1/100 秒

注 1 Solaris の場合の computerSystemFreeMemory についての注意事項を次に示します。

#### 取得時間

Solaris の computerSystemFreeMemory の値は取得するために 6 秒以上必要です。取得する場合はマネージャーシステムの SNMP 要求のタイムアウト時間を 6 秒以上に指定してください。

#### Solaris の物理メモリー

Solaris の物理メモリーは、通常のプログラムの動作などに使用される以外に、ファイルキャッシュ（バッファキャッシュ）としても使用されています。ファイルキャッシュとして使用されている部分で再利用できる物理メモリーについては、すぐにメモリーが解放されません。再び同一ファイルが参照されるときはキャッシュとして残されています。したがって、システムを一定時間継続して動作させると、computerSystemFreeMemory の値は、ほぼ一定の値に収束してしまいます（具体的な値は、システムごとに異なります）。このため、Solaris では、いわゆるシステムの空きメモリーやメモリー使用率を監視する場

#### 4. MIB オブジェクト一覧

合には、物理メモリーではなく仮想メモリー（スワップ）を監視することを推奨します。仮想メモリー（スワップ）の監視には、`computerSystemSwapConfig` や `computerSystemFreeSwap` が使用できます。`computerSystemFreeMemory` は、ファイルキャッシュ分も含めた物理メモリーの空き容量を監視する場合に使用してください。

注 2 各 OS の `computerSystemSwapConfig` および `computerSystemFreeSwap` に物理メモリーが含まれているかどうかを、次に示します。

HP-UX (IPF) , Linux , Linux (IPF) の場合

物理メモリーは含まれません。

Solaris の場合

物理メモリーは含まれます。

AIX の場合

AIX については、実際にはページングスペースの使用状況を応答していますので注意してください。物理メモリーは含まれません。

注 3 Solaris の場合のスワップ空間についての注意事項を次に示します。

Solaris のスワップ空間はディスク上のスワップ領域に未使用の実メモリーが含まれ、実メモリーでの仮想記憶領域は動的に確保されます。そのため、`computerSystemSwapConfig` の値は動的に変化します。

注 4 AIX の場合、単位はバイトです。

注 5 CPU 情報についての注意事項を次に示します。

- Solaris および AIX では、SNMP エージェントに設定されている CPU 利用時間情報取得のインターバル時間（デフォルト 5 分）ごとに、CPU 情報を更新します。そのため、CPU 利用時間を収集する場合は、CPU 利用情報取得のインターバル時間より大きい収集間隔を設定してください。  
この CPU 利用時間情報取得のインターバル時間（単位分）は、CPU 利用時間を定期的に収集するデーモンプロセスである `htc_monagt1` プロセスの `-s` オプションで設定します。このインターバル時間の範囲は、0 から 1440 です。なお、0 の場合は CPU 利用時間情報を取得しません。また、Solaris および AIX の場合、SNMP エージェントが起動してから最初の CPU 利用時間情報を取得するまでのインターバル時間内では、CPU 利用時間情報すべての MIB 値は、`noSuchName` で応答します。
- Solaris では、CPU の `online/offline` 状態を変える場合があります。CPU の状態が変更された場合で、CPU 利用情報取得のインターバル時間内のとき、すべての MIB 値は `noSuchName` で応答します。しかし、CPU 利用情報取得のインターバル時間が経過すれば、すべての MIB 値は取得できるようになり、CPU 状態の変更を実行したあとの CPU 利用情報取得のインターバル時間から、値がリセットされます。ただし、CPU の状態が変更されても、取得する CPU 番号は変わりません。

- AIX では、DLPAR (Dynamic Logical Partition) によって CPU がダイナミックに追加・削除される場合があります。CPU がダイナミックに追加・削除された場合で、CPU 利用情報取得のインターバル時間内のとき、すべての MIB 値は noSuchName で応答します。しかし、CPU 利用情報取得のインターバル時間が経過すれば、すべての MIB 値は取得できるようになり、CPU の追加・削除を実行したあとの CPU 利用情報取得のインターバル時間から、値がリセットされません。

注 6 使用中の物理メモリー容量の取得についての注意事項を次に示します。

AIX では使用中の物理メモリーをファイルキャッシュとして利用することで、ファイルアクセスを向上させています。そのため、SNMP エージェントが取得する使用中の物理メモリー容量には、デフォルトでファイルキャッシュ値が含まれています。環境変数定義ファイル SnmpHpunix ファイルで SNMP\_HTC\_AIX\_EXCEPT\_FILECACHE 環境変数に Y を指定すると、ファイルキャッシュ値を含まない使用中の物理メモリー容量を取得できます。環境変数定義ファイルのパスは、「付録 A SNMP エージェントのファイルの一覧」を参照してください。

SNMP\_HTC\_AIX\_EXCEPT\_FILECACHE 環境変数の指定例を次に示します。

(例)

```
SNMP_HTC_AIX_EXCEPT_FILECACHE=Y
export SNMP_HTC_AIX_EXCEPT_FILECACHE
```

## (2) FileSystem グループ

FileSystem グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-13 FileSystem グループ (enterprises.hp.nm.system.general.fileSystem)  
(1.11.2.3.1.2)

ID	オブジェクト名	内容	単位
1	fileSystemMounted	マウントされているファイルシステムの数。	-
2	fileSystemTable	ファイルシステム情報テーブル。	-
2.1	fileSystemEntry	各エントリーは fileSystemID1 と fileSystemID2 の値で識別される。	-
2.1.1	fileSystemID1	ファイルシステム ID。	-
2.1.2	fileSystemID2	ファイルシステム ID (ファイルサーバのタイプなどの情報)。	-
2.1.3	fileSystemName	マウントされているファイルシステム名。	-
2.1.4	fileSystemBlock	ファイルシステム中のトータルブロック数。	ブロック
2.1.5	fileSystemBfree	ファイルシステム中のフリーブロック数。	ブロック
2.1.6	fileSystemBavail	スーパーユーザーではない場合、利用できるフリーブロック数。	ブロック

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.7	fileSystemBsize	基本ファイルシステムのブロックサイズ。	バイト
2.1.8	fileSystemFiles	ファイルシステム中の i ノード総数。	-
2.1.9	fileSystemFfree	ファイルシステム中の未使用 i ノード総数。	-
2.1.10	fileSystemDir	ファイルシステムの path。	-

#### 注意事項

FileSystem グループの HP 企業固有 MIB オブジェクトの注意事項を次に示します。

- AIX, Linux および Linux(IPF) では, fileSystem グループについても fileSystem64 グループと同様に, /etc/SnmpAgent.d/esafilesys.conf の設定が有効になります。fileSystem64 グループの /etc/SnmpAgent.d/esafilesys.conf の設定については, 「4.3.2 日立企業固有 MIB オブジェクトの内容」の「(20) fileSystem64 グループ」を参照してください。  
Solaris および HP-UX(IPF) では, /etc/SnmpAgent.d/esafilesys.conf の設定が有効にならないのでご注意ください。
- AIX 5L でファイルシステムが JFS2 の場合は, 最大 4 ペタバイトのファイルシステムを構築できますが, SNMP エージェントの hp.nm.system.general.fileSystem グループの MIB で表現できるファイルシステムのサイズの最大値を次に示します。i ノード数は, 最大  $2^{31}-1$  まで表現できます。

ファイルシステムのブロックサイズ(単位: バイト)	ファイルシステムのサイズ(単位: テラバイト)
512	1
1024	2
4096	8

- Linux, Linux (IPF) では, hp.nm.system.general.fileSystem グループのファイルシステム情報にある「マウントされているファイルシステム名」が, df コマンドの表示と異なります。これは, このソフトウェア製品が /etc/fstab の情報を参照しているからです。/etc/fstab の内容とは一致しています。次に例を示します。

(例)

MIB の情報

```
hp.nm.system.general.fileSystem.fileSystemTable.fileSystemEntry.fileSystemName.1.1
: DISPLAY STRING- (ascii): LABEL=/
```

/etc/fstab の情報

```
LABEL=/          /          ext2  defaults  1 1
```

## df の情報

```
df
Filesystem      1k-blocks      Used Available Use% Mounted on
/dev/hda1       6048320       727156   5013924  13% /
```

- Solaris の hp.nm.system.general.fileSystem グループは、tmpfs ファイルシステム形式のファイルシステムを MIB 値に追加していません。そのため、tmpfs ファイルシステム形式のファイルシステムを監視できません。

## (3) Processes グループ

Processes グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-14 Processes グループ ( enterprises.hp.nm.system.general.processes )  
( 1.11.2.3.1.4 )

ID	オブジェクト名	内容	単位
1	processNum	実行中のプロセス数。	-
2	processTable	実行中のプロセス情報テーブル。	-
2.1	processEntry	各エントリーは processPID の値で識別される。	-
2.1.1	processPID	プロセス ID。	-
2.1.2	processIdx	プロセスのエントリー番号。	-
2.1.3	processUID	プロセスのユーザー ID。	-
2.1.4	processPPID	親プロセス ID。	-
2.1.5	processDsize	プロセスのデータ領域サイズ。	ページ
2.1.6	processTsize	プロセスのテキスト領域サイズ。	ページ
2.1.7	processSsize	プロセスのスタック領域サイズ。	ページ
2.1.8	processNice <sup>1</sup>	プロセスのナイス値。	-
2.1.9	processMajor	プロセスの制御端末のメジャー番号。	-
2.1.10	processMinor	プロセスの制御端末のマイナー番号。	-
2.1.11	processPgrp	プロセスグループ ID。	-
2.1.12	processPrio	プロセスの優先度。	-
2.1.13	processAddr	プロセスのユーザー領域アドレス。値は、プロセスがメモリー上にロードされている場合はメモリー中の物理アドレス、スワップアウトされている場合はディスク上のアドレスを示す。	-
2.1.14	processCPU	スケジューリングのためのプロセッサ利用率。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.15	processUtime	プロセスがユーザーモードで実行した経過時間。	1/100 秒
2.1.16	processStime	プロセスがシステム（カーネル）モードで実行した経過時間。	1/100 秒
2.1.17	processStart	プロセスが開始してからの経過時間。	秒
2.1.18	processFlags <sup>2</sup>	プロセスフラグ。 incore ( 1 ), sys ( 2 ), locked ( 4 ), trace ( 8 ), trace2 ( 16 )	-
2.1.19	processStatus	プロセスの実行状態。 sleep ( 1 ), run ( 2 ), stop ( 3 ), zombie ( 4 ), other ( 5 ), idle ( 6 )	-
2.1.20	processWchan	プロセスがスリープしているアドレス。	-
2.1.21	processProcNum	プロセスを最後に実行したプロセス。	-
2.1.22	processCmd	プロセスを実行したコマンドライン文字列。	-
2.1.23	processTime	プロセスのインコアでの経過時間。	秒
2.1.24	processCPUticks	現在のタイムスライス中にプロセスが消費した CPU のティック数。	-
2.1.25	processCPUticksTotal <sup>3</sup>	プロセス生成以降、プロセスが消費した CPU のティック数。	-
2.1.26	processFss	プロセスが所属する均等割当てスケジューラグループの ID。	-
2.1.27	processPctCPU	インコアでの経過時間に占める CPU 使用時間の割合。	%
2.1.28	processRssize	インコアにロードされているページ数。	ページ
2.1.29	processSUID	プロセスの実ユーザー ID。	-
2.1.30	processUname	プロセスを起動したユーザー名。	-
2.1.31	processTTY	プロセスの tty。	-

注 1

Linux および Linux (IPF) では、通常 nice の値は - 20 から 19 の間の値を取ります。processNice の MIB 値は Gauge で取得される必要があり、負の値に対応していません。したがって、取得した nice の値に 20 を加え 0 から 39 の間の値に変換したものを processNice の MIB 値とします。

注 2

Solaris の場合、process.processTable.processEntry.processFlag の値には意味がありません。

注 3

Linux (IPF) の場合、processCPUticksTotal の値は 497 日でラップアラウンドします。

#### (4) Cluster グループ

Cluster グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-15 Cluster グループ (enterprises.hp.nm.system.general.cluster) (1.11.2.3.1.5)

ID	オブジェクト名	内容	単位
1	isClustered	マシンがクラスタシステムかどうかの識別。 standalone (1), rootserver (2), cnode (3)	-
2	clusterTable	クラスタノード情報テーブル。	-
2.1	clusterEntry	各エントリは clusterID の値で識別される。	-
2.1.1	clusterID	cnode ID。	-
2.1.2	clusterMachineID	cnode マシン ID。	-
2.1.3	clusterType	cnode タイプ (r または c)。	-
2.1.4	clusterCnodeName	cnode 名。	-
2.1.5	clusterSwapServingCnode	スワップサービング cnode。	-
2.1.6	clusterKcsp	KCSP。	-
2.1.7	clusterCnodeAddress	cnode IP アドレス。	-
3	clusterCnodeID	マシンの cnode ID。	-

#### (5) ieee8023Mac グループ

ieee8023Mac グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-16 ieee8023Mac グループ (enterprises.hp.nm.interface.ieee8023Mac) (1.11.2.4.1)

ID	オブジェクト名	内容	単位
1	ieee8023MacTable	ieee802.3 インタフェース情報テーブル。	-
1.1	ieee8023MacEntry	各エントリは ieee8023MacIndex の値で識別される。	-
1.1.1	ieee8023MacIndex	ifIndex に対応したインタフェース番号。	-
1.1.2	ieee8023MacTransmitted	転送に成功したフレーム数。	フレーム
1.1.3	ieee8023MacNot Transmitted	転送されなかったフレーム数。	フレーム
1.1.4	ieee8023MacDeferred	媒体の busy によって据え置かれたフレーム数。	フレーム
1.1.5	ieee8023MacCollisions	衝突のために再送された転送試行回数の総称。	-
1.1.6	ieee8023MacSingle Collisions	一つの衝突に妨げられ、そのあと転送に成功した転送試行回数。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
1.1.7	ieee8023MacMultiple Collisions	2 ~ 5 回の衝突試行で妨げられ、そのあと転送に成功した転送試行回数。	-
1.1.8	ieee8023MacExcess Collisions	15 回以上衝突試行で妨げられ、そのあと転送に成功した転送試行回数。	-
1.1.9	ieee8023MacLateCollisions	割り当てられたチャンネル時間が経過したあと、発生した衝突のために失敗した転送試行回数。	-
1.1.10	ieee8023MacCarrierLost Errors	転送を試みたとき、carrier sense が失われた回数。	-
1.1.11	ieee8023MacNoHeartBeat Errors	転送のあと、heart beat が指し示されることのない回数。	-
1.1.12	ieee8023MacFrames Received	受信に成功したフレーム数。	フレーム
1.1.13	ieee8023MacUndeliverable FramesReceived	フレームが受信されるより早く送信されたとき、ソフトウェアバッファがオーバーランのため配送されなかった受信フレーム数。	フレーム
1.1.14	ieee8023MacCRCErrors	CRC エラー発見回数。	-
1.1.15	ieee8023MacAlignment Errors	misaligned で、誤った CRC 両方の受信フレーム数。	フレーム
1.1.16	ieee8023MacResource Errors	リソースが不足したため、失われた受信フレーム数。	フレーム
1.1.17	ieee8023MacControlField Errors	コントロールフィールドにエラーがある受信フレーム数。	フレーム
1.1.18	ieee8023MacUnknown ProtocolErrors	タイプフィールドまたはサブフィールドが誤ったプロトコルを参照したためドロップしたフレーム数。	フレーム
1.1.19	ieee8023MacMulticasts Accepted	マルチキャストアドレスを受け取った数。	-

#### (6) ICMP グループ

ICMP グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-17 ICMP グループ ( enterprises.hp.nm.icmp )( 1.11.2.7 )

ID	オブジェクト名	内容	単位
1	icmpEchoReq	ICMP エコー要求に回答するために掛かる秒数およびエラー情報。	-

#### 注

SNMP エージェントをインストールしているホストから指定されたホストへ、ICMP エコーリクエストを送信します。その応答に掛かる時間 ( ミリ秒 ) を MIB 値にします。ICMP エコーリクエストでエラーが発生した場合は次の値になります。

-1 : 内部エラー発生

- 2: ICMP エコーリクエストがタイムアウト
- 3: エコーリプライが正しくない
- 4: パケットサイズが大き過ぎる
- 5: タイムアウト値が間違っている

この MIB 値は SNMP-GET 要求でだけ取得できます。SNMP GET-NEXT 要求では取得できません。SNMP-GET 要求では SNMP エージェントをインストールしているホストから、ICMP エコーリクエストを送信するホストの IP アドレス、その ICMP エコーリクエストのパケットサイズ (バイト)、および ICMP エコーリクエストのタイムアウト時間 (秒) を指定します。IP アドレスを a1.a2.a3.a4、パケットサイズを s、タイムアウト時間を t とした場合、リクエストの形式は icmpEchoReq.s.t.a1.a2.a3.a4 となります。

次に例を示します。

(例)

IP アドレス 15.2.112.113 に、タイムアウトを 8 秒、パケットサイズを 75 バイトで、ICMP エコーリクエストを送りたい場合は、次のように指定します。

```
icmpEchoReq.75.8.15.2.112.113
```

## (7) Trap グループ

Trap グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-18 Trap グループ ( enterprises.hp.nm.snmp.trap ) ( 1.11.2.13.1 )

ID	オブジェクト名	内容	単位
1	trapDestinationNum	トラップあて先の数。	-
2	trapDestinationTable	エージェントのトラップあて先アドレステーブル。	-
2.1	trapDestinationEntry	各エントリーは trapDestination の値で識別される。	-
2.1.1	trapDestination	エージェントのトラップあて先アドレス。	-

## (8) SnmpdConf グループ

SnmpdConf グループの HP 企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-19 SnmpdConf グループ ( enterprises.hp.nm.snmp.snmpdConf ) ( 1.11.2.13.2 )

ID	オブジェクト名	内容	単位
1	snmpdConfRespond	true (1) がセットされていると全オブジェクトに応答する。 true (1), false (2)	-
2	snmpdReConfigure	reset (1) がセットされるとエージェントが再構成される。	-
3	snmpdFlag	エージェントの能力。 removetrap (1), netwareproxy (2)	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
4	snmpdLogMask	エージェントのログマスク値。	-
5	snmpdVersion	エージェントのバージョン番号。	-
6	snmpdStatus	デーモンの状態。 up (1), down (2)	-
7	snmpdSize	エージェントのデータセグメントの大きさ。	バイト
9	snmpdWhatString	エージェントのプロフィール。 製品名, バージョン, 日付, コピーライト	-

### 4.2.3 HP 企業固有 MIB オブジェクトの実装状況

HP 企業固有 MIB オブジェクトの実装状況を表 4-20 ~ 表 4-27 に示します。なお, 表中では次の凡例を使用しています。

(凡例)

: get オペレーションで値を取得できる。または set オペレーションで値をセットできる。

N : get オペレーションで値を取得できない。または set オペレーションで値をセットできない。noSuchName で応答する。

F (値):(値) で示す固定値で応答する。

- : アクセス権限がない。noSuchName で応答する。

#### (1) ComputerSystem グループ

ComputerSystem グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-20 HP 企業固有 MIB オブジェクトの実装状況 (computerSystem グループ)  
(enterprises.hp.nm.system.general.computerSystem)(1.11.2.3.1.1)

オブジェクト名	MIB オペレーション									
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux の場合		Linux (IPF) の場合	
	get	set	get	set	get	set	get	set	get	set
computerSystemUpTime		-		-		-		-		-
computerSystemUsers		-		-		-		-		-
computerSystemAvgJobs1		-		-		-		-		-
computerSystemAvgJobs5		-		-		-		-		-
computerSystemAvgJobs15		-		-		-		-		-
computerSystemMaxProc		-	N	-	-	-	N	-	N	-

オブジェクト名	MIB オペレーション									
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux の場合		Linux (IPF) の場合	
	get	set	get	set	get	set	get	set	get	set
computerSystemFreeMemory		-		-		-		-		-
computerSystemPhysMemory		-		-		-		-		-
computerSystemMaxUserMemory		-	N	-	-	-	N	-	N	-
computerSystemSwapConfig		-		-		-		-		-
computerSystemEnabledSwap		-	N	-	-	-	N	-	N	-
computerSystemFreeSwap		-		-		-		-		-
computerSystemUserCPU		-		-		-		-	N	-
computerSystemSysCPU		-		-		-		-	N	-
computerSystemIdleCPU		-		-		-		-	N	-
computerSystemNiceCPU		-	N	-	-	-		-	N	-

## (2) FileSystem グループ

FileSystem グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-21 HP 企業固有 MIB オブジェクトの実装状況 (fileSystem グループ)  
(enterprises.hp.nm.system.general.fileSystem)(1.11.2.3.1.2)

オブジェクト名	MIB オペレーション									
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux の場合		Linux (IPF) の場合	
	get	set	get	set	get	set	get	set	get	set
fileSystemMounted		-		-		-		-		-

#### 4. MIB オブジェクト一覧

オブジェクト名		MIB オペレーション									
		HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux の場合		Linux (IPF) の場合	
		get	set	get	set	get	set	get	set	get	set
file System Table. file System Entry. ~	fileSystemID1		-		-		-		-		-
	fileSystemID2		-		-		-		-		-
	fileSystemName		-		-		-		-		-
	fileSystemBlock		-		-		-		-		-
	fileSystemBfree		-		-		-		-		-
	fileSystemBavail		-		-		-		-		-
	fileSystemBsize		-		-		-		-		-
	fileSystemFiles		-		-		-		-	N	-
	fileSystemFfree		-		-		-		-	N	-
	fileSystemDir		-		-		-		-		-

### (3) Processes グループ

Processes グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-22 HP 企業固有 MIB オブジェクトの実装状況 (processes グループ)  
(enterprises.hp.nm.system.general.processes) (1.11.2.3.1.4)

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
processNum			-		-		-		-
process Table. process Entry. ~	processPID		-		-		-		-
	processIdx		-	N	-		-	N	-
	processUID		-		-		-		-
	processPPID		-		-		-		-
	processDsize		-	N	-		-	N	-
	processTsize		-	N	-		-	N	-
	processSsize		-	N	-		-	N	-
processNice		-		-		-		-	

オブジェクト名	MIB オペレーション							
	HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
	get	set	get	set	get	set	get	set
processMajor		-	N	-	-	-	N	-
processMinor		-	N	-	-	-	N	-
processPgrp		-		-	-	-		-
processPrio		-		-		-		-
processAddr		-	N	-	-	-	N	-
processCPU		-		-	-	-	N	-
processUtime		-		-	-	-	N	-
processStime		-		-	-	-	N	-
processStart		-		-	-	-	N	-
processFlags		-		-	-	-	N	-
processStatus		-		-	-	-	N	-
processWchan		-		-	-	-	N	-
processProcNum		-		-	-	-	N	-
processCmd		-		-		-		-
processTime		-	F (0)	-	-	-	N	-
processCPU ticks		-	N	-	-	-	N	-
processCPU ticksTotal		-		-		-		-
processFss		-	N	-	-	-	N	-
processPctCPU		-	N	-	-	-	N	-
processRssize		-		-	-	-	N	-
processSUID		-		-	-	-	N	-
processUname		-		-		-		-
processTTY	N	-	N	-	N	-	N	-

#### 4. MIB オブジェクト一覧

##### (4) Cluster グループ

Cluster グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-23 HP 企業固有 MIB オブジェクトの実装状況 (cluster グループ)  
(enterprises.hp.nm.system.general.cluster)(1.11.2.3.1.5)

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
isClusterd			-	N	-	-	-	-	-
cluster Table. cluster Entry. ~	clusterID		-	N	-	-	-	-	-
	ClusterMachineID		-	N	-	-	-	-	-
	clusterType		-	N	-	-	-	-	-
	clusterCnodeName		-	N	-	-	-	-	-
	clusterSwapServingCnode		-	N	-	-	-	-	-
	clusterKcsp		-	N	-	-	-	-	-
	clusterCnodeAddress		-	N	-	-	-	-	-
clusterCnodeID			-	N	-	-	-	-	-

##### (5) Ieee8023Mac グループ

Ieee8023Mac グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-24 HP 企業固有 MIB オブジェクトの実装状況 ( ieee8023Mac グループ )  
 ( enterprises.hp.nm.interface.ieee8023Mac ) ( 1.11.2.4.1 )

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
ieee8023 Mac Table. ieee8023 Mac Entry. ~	ieee8023Mac Index		-	N	-	-	-	-	-
	ieee8023Mac Transmitted		-	N	-	-	-	-	-
	ieee8023Mac Not Transmitted		-	N	-	-	-	-	-
	ieee8023Mac Deferred		-	N	-	-	-	-	-
	ieee8023Mac Collisions		-	N	-	-	-	-	-
	ieee8023Mac Single Collisions		-	N	-	-	-	-	-
	ieee8023Mac Multiple Collisions		-	N	-	-	-	-	-
	ieee8023Mac Excess Collisions		-	N	-	-	-	-	-
	ieee8023Mac LateCollisions		-	N	-	-	-	-	-
	ieee8023Mac CarrierLost Errors		-	N	-	-	-	-	-
	ieee8023Mac NoHeartBeat Errors		-	N	-	-	-	-	-
	ieee8023Mac Frames Received		-	N	-	-	-	-	-

#### 4. MIB オブジェクト一覧

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
	ieee8023Mac Undeliverable FramesReceived		-	N	-	-	-	-	-
	ieee8023Mac CRCErrors		-	N	-	-	-	-	-
	ieee8023Mac Alignment Errors		-	N	-	-	-	-	-
	ieee8023Mac Resource Errors		-	N	-	-	-	-	-
	ieee8023Mac ControlField Errors		-	N	-	-	-	-	-
	ieee8023Mac Unknown Protocol Errors		-	N	-	-	-	-	-
	ieee8023Mac Multicasts Accepted		-	N	-	-	-	-	-

#### (6) ICMP グループ

ICMP グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-25 HP 企業固有 MIB オブジェクトの実装状況 (ICMP グループ)  
( enterprises.hp.nm.icmp )( 1.11.2.7 )

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
	icmpEchoReq		-		-		-		-

#### (7) Trap グループ

Trap グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-26 HP 企業固有 MIB オブジェクトの実装状況 ( trap グループ )  
( enterprises.hp.nm.snmp.trap )( 1.11.2.13.1 )

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
trapDestinationNum			-		-		-		-
trapDestination Table. trapDestination Entry. ~	trapDestination								

#### ( 8 ) SnmpdConf グループ

SnmpdConf グループの HP 企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-27 HP 企業固有 MIB オブジェクトの実装状況 ( snmpdConf グループ )  
( enterprises.hp.nm.snmp.snmpdConf )( 1.11.2.13.2 )

オブジェクト名		MIB オペレーション							
		HP-UX (IPF)		Solaris		AIX		Linux , Linux (IPF)	
		get	set	get	set	get	set	get	set
snmpdConfRespond									
snmpdReConfigure									
snmpdFlag			-		-		-		-
snmpdLogMask									
snmpdVersion			-		-		-		-
snmpdStatus									
snmpdSize			-		-		-		-
snmpdWhatString			-		-		-		-

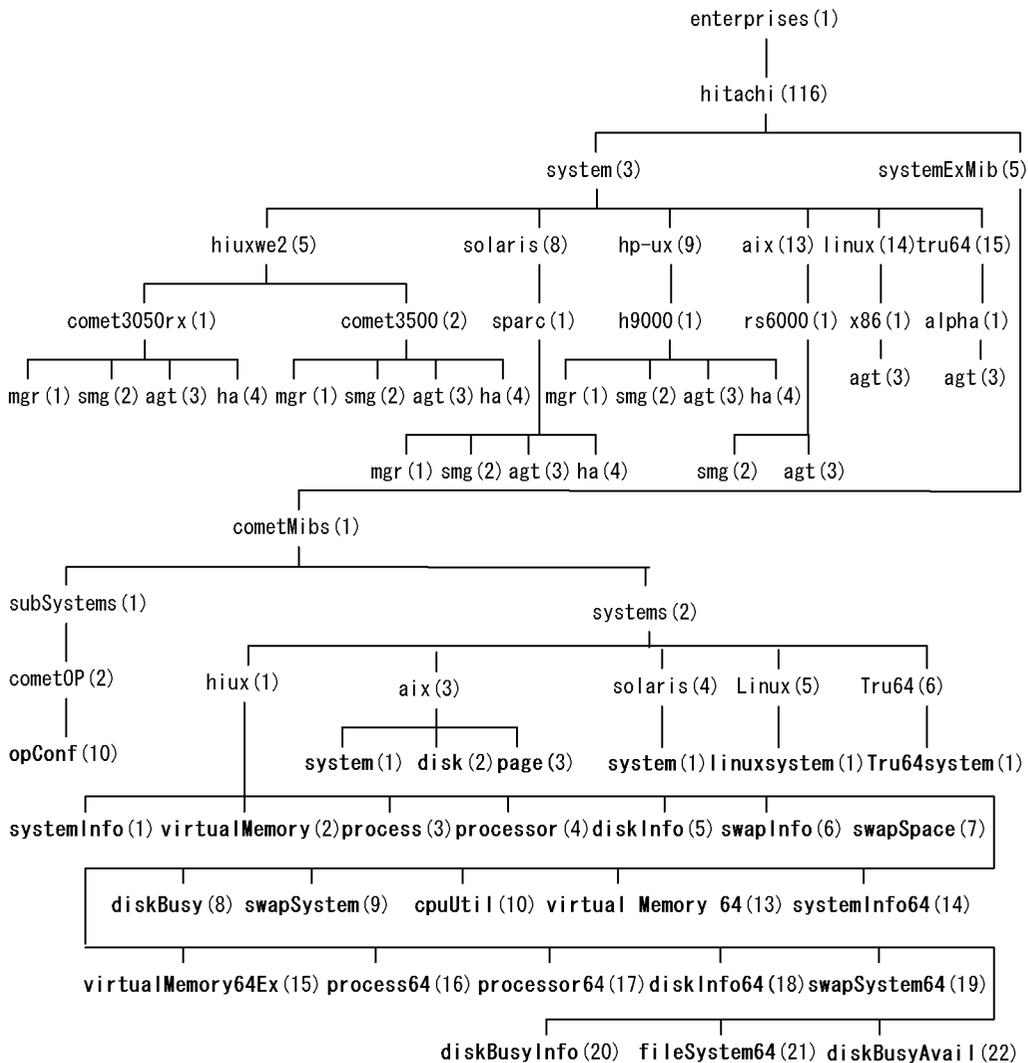
## 4.3 日立企業固有 MIB オブジェクト一覧

この節では、SNMP エージェントが実装している日立企業固有 MIB オブジェクトの一覧と実装状況について説明します。

### 4.3.1 日立企業固有 MIB オブジェクトの構成

日立企業固有 MIB オブジェクトの構成を次の図に示します。

図 4-3 日立企業固有 MIB オブジェクトの構成



(凡例) 太字：この節では、太字の日立企業固有MIBオブジェクトを説明しています。

日立企業固有 MIB オブジェクトの内容および実装状況の参照先一覧を次の表に示します。

表 4-28 日立企業固有 MIB オブジェクトの参照先一覧

日立企業固有 MIB オブジェクトのグループ				参照先	
				MIB オブジェクトの内容	MIB オブジェクトの実装状況
enterprises.hitachi.systemExMib. CometMibs	SubSystems	cometOP	OpConf	表 4-29	表 4-57
	systems	hiux	systemInfo	表 4-30	表 4-58
			virtualMemory	表 4-31	表 4-59
			process	表 4-32	表 4-60
			processor	表 4-33	表 4-61
			diskInfo	表 4-34	表 4-62
			swapInfo	表 4-35	表 4-63
			swapSpace	表 4-36	表 4-64
			diskBusy	表 4-37	表 4-65
			SwapSystem	表 4-38	表 4-66
			cpuUtil	表 4-39	表 4-67
			virtualMemory64	表 4-40	表 4-68
			systemInfo64	表 4-41	表 4-69
			virtualMemory64 Ex	表 4-42	表 4-70
			process64	表 4-43	表 4-71
			processor64	表 4-44	表 4-72
			diskInfo64	表 4-45	表 4-73
			swapSystem64	表 4-46	表 4-74
			diskBusyInfo	表 4-47	表 4-75
			fileSystem64	表 4-48	表 4-76
			diskBusyAvail	表 4-49	表 4-77
	aix <sup>1</sup>	system	表 4-50	表 4-78	
		disk	表 4-51		
		page	表 4-52		
	solaris <sup>2</sup>	system	表 4-53	表 4-79	

#### 4. MIB オブジェクト一覧

日立企業固有 MIB オブジェクトのグループ				参照先	
				MIB オブジェクトの内容	MIB オブジェクトの実装状況
	Linux	3	linuxsystem	表 4-54	表 4-80
	Tru64	4	Tru64system	表 4-55	表 4-81

- 注 1 AIX システム固有の MIB オブジェクトのグループです。  
 注 2 Solaris システム固有の MIB オブジェクトのグループです。  
 注 3 Linux システム固有の MIB オブジェクトのグループです。  
 注 4 Tru64 システム固有の MIB オブジェクトのグループです。

### 4.3.2 日立企業固有 MIB オブジェクトの内容

グループごとの日立企業固有 MIB オブジェクトの内容を、表 4-29 ~ 表 4-55 に示します。なお、表中では次の凡例を使用しています。

(凡例)

- : 該当なし。

MIB オブジェクトの内容は、`/var/opt/OV/share/snmp_mibs/eagent` 配下の次のファイルでも参照できます。

- hitachi-cometAgt
- hitachi-cometAgt-solaris
- hitachi-cometAgt-aix
- hitachi-cometAgt-linux
- hitachi-cometAgt-tru64

#### (1) OpConf グループ

OpConf グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-29 OpConf グループ

(enterprises.hitachi.systemExMib.CometMibs.SubSystems.cometOP.OpConf)  
 (1.116.5.1.1.2.10)

ID	オブジェクト名	内容	単位
1	OpConfCharCode	操作支援がコード変換に使用する文字コード。	-

#### (2) SystemInfo グループ

SystemInfo グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-30 SystemInfo グループ  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.systemInfo )  
 ( 1.116.5.1.2.1.1 )

ID	オブジェクト名	内容	単位
1	systemInfoTTYMajor	制御端末のメジャー番号。	-
2	systemInfoTTYMinor	制御端末のマイナー番号。	-
3	systemInfoBootTime	システムがブートしてからの経過時間。	1/100 秒
4	systemInfoActiveProcessors	システムコールを発行した時点で動作しているプロセッサ数。	-
5	systemInfoProcessors	システムがブートされてから今までに同時に動作したことがある最大のプロセッサの数。	-
6	systemInfoMaxProcesses	システムが動作させることのできる最大のプロセス数。	-
7	systemInfoRunQue Processes	システムプロセス以外の実行待ち状態のプロセスで、主記憶装置上にロードされているプロセスの数。	-
8	systemInfoXferWait Processes	ディスクからのデータ転送を待っているプロセス数。	-
9	systemInfoPageInWait Processes	ディスクからページインされるのを待っているプロセス数。	-
10	systemInfoSleepProcesses	物理メモリー中でスリープしているプロセス数。ただし、次の状態のプロセスは含まれない。 <ul style="list-style-type: none"> <li>• I/O 待ちでスリープしているプロセス</li> <li>• 連続して 20 秒以上スリープしているプロセス</li> </ul>	-
11	systemInfoSwapOut Processes	スワップアウトされているプロセス数。ただし、スワップインされたらすぐに実行待ち状態になるプロセスだけである。	-
12	systemInfoPhysicalMemory Size	システムにインストールされている最大物理メモリーのサイズ。	バイト
13	systemInfoPhysicalMemory FreeSize	未使用でかつ、システムがプロセスに割り当てできるとみなした物理メモリーのサイズ。	バイト
14	systemInfoVirtualMemory ProcessSize	システムプロセス以外のすべてのプロセスのテキスト、データ、スタックとして割り当てられている仮想メモリーのサイズ。	バイト
15	systemInfoVirtualMemoryWaitPr ocessSize	実行待ち状態のすべてのプロセスに割り当てられている仮想メモリーのサイズ。	バイト
16	systemInfoPhysicalMemory ProcessSize	システム内にあるすべてのプロセスの、テキスト、データ、スタックに割り当てられている物理メモリーのサイズ。	バイト
17	systemInfoPhysicalMemory WaitProcessSize	実行待ち状態のすべてのプロセスに割り当てられている物理メモリーのサイズ。	バイト
18	systemInfoCPUStates	システムが持つ CPU 状態の数。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
19	systemInfoOpenLogical Volumes	オープン済みの論理ボリュームの数。	-
20	systemInfoOpenLogical VolumeGrps	オープン済みの論理ボリュームグループの数。	-
21	systemInfoAllocPBUFs	論理ボリューム用に割り当てられている pbuf の数。	-
22	systemInfoUsedPBUFs	論理ボリュームで使用している pbuf の数。	-
23	systemInfoMaxPBUFs	今までに論理ボリュームによって使用された pbuf の最大数。	-
24	systemInfoActiveProcess Entries	プロセステーブルの総エントリー数の中で現在使用中のエントリー数。	エントリー
25	systemInfoActiveInode Entries	i ノードテーブルの総エントリー数の中で現在使用中のエントリー数。	エントリー
26	systemInfoActiveFile Entries	ファイルテーブルの総エントリー数の中で現在使用中のエントリー数。	エントリー

### (3) VirtualMemory グループ

VirtualMemory グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-31 VirtualMemory グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory )  
( 1.116.5.1.2.1.2 )

ID	オブジェクト名	内容	単位
1	vmPageSize	仮想記憶システムでの 1 ページのバイト数。	バイト
2	vmDaemonFreePages	ページアウトデーモンにフリーにされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
3	vmInterruptions	デバイスからの ( ハードウェア ) 割り込み回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
4	vmPageInPages	ページインされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
5	vmPageOutPages	ページアウトされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
6	vmPageReclaims	ページリクレームの合計数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
7	vmTLBFlashes	TLB フラッシュの回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
8	vmDaemonScanPages	ページアウトデーモンにスキャンされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ

ID	オブジェクト名	内容	単位
9	vmContextSwitches	コンテキストスイッチの回数。この値は1秒当たりの平均値で、5秒ごとに更新される。	-
10	vmSystemCalls	システムコールの発行回数。この値は1秒当たりの平均値で、5秒ごとに更新される。	-
11	vmXFileSystemFreelist Pages	ファイルシステムからではなく、ページフリーリストから見つかったページ数。この値は1秒当たりの平均値で、5秒ごとに更新される。	ページ
12	vmXSwapDeviceFreeList Pages	スワップデバイスからではなく、ページフリーリストから見つかったページ数。この値は1秒当たりの平均値で、5秒ごとに更新される。	ページ
13	vmFreeMemoryPages	フリーメモリーページ数。この値は1秒当たりの平均値で、5秒ごとに更新される。	ページ
14	vmTotalSwapIns	ブート以降のスワップインの回数。	-
15	vmTotalSwapOuts	ブート以降のスワップアウトの回数。	-
16	vmTotalDaemonFreePages	ブート以降にページアウトデーモンにフリーにされたページ数。	ページ
17	vmTotalDemandLoadPages	ブート以降に、実行可能ファイルにデマンドロードされたページ数。	ページ
18	vmTotalPageFaults	ブート以降のページフォルトの回数。	-
19	vmTotalInterruptions	ブート以降のデバイス（ハードウェア）割り込みの回数。	-
20	vmTotalIntransitPage Faults	ブート以降の intransit ブロックページフォルトの回数。	-
21	vmTotalDemandLoadCreate Pages	ブート以降に、新たに作成されたデマンドロードページ数。	ページ
22	vmTotalZeroFillCreate Pages	ブート以降に、新たに作成されたゼロクリアページ数。	ページ
23	vmTotalFreeListReclaimed Pages	ブート以降に、フリーリストから再利用されたページ数。	ページ
24	vmTotalPageIns	ブート以降のページインの回数。	-
25	vmTotalPageOuts	ブート以降のページアウトの回数。	-
26	vmTotalPageInPages	ブート以降にページインされたページ数。	ページ
27	vmTotalPageOutPages	ブート以降にページアウトされたページ数。	ページ
28	vmTotalSwapInPages	ブート以降にスワップインされたページ数。	ページ
29	vmTotalSwapOutPages	ブート以降にスワップアウトされたページ数。	ページ
30	vmTotalDaemonTicksNum	ブート以降にページアウトデーモンがページアウトした回数。	-
31	vmTotalContextSwitches	ブート以降のコンテキストスイッチの回数。	-
32	vmTotalSystemCalls	ブート以降に発行されたシステムコールの回数。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
33	vmTotalTraps	ブート以降に発生したトラップの回数。	-
34	vmTotalXFileSystemFreeListPages	ブート以降にファイルシステムからではなくフリーリストから見つかったページ数。	ページ
35	vmTotalXSwapDeviceFreeListPages	ブート以降にスワップデバイスからではなくフリーリストから見つかったページ数。	ページ
36	vmTotalDemandZeroFillPages	ブート以降にオンデマンドでゼロクリアされたページ数。	ページ
37	vmTotalDaemonScanPages	ブート以降にページアウトデーモンにスキャンされたページ数。	ページ
38	vmTotalReclaimedPages	ブート以降に再利用されたページ数。	ページ
39	vmTotalDeficitPages	新たにスワップインされようとしているプロセスに必要なと思われるページ数。	ページ
40	vmTotalReadChars	ブート以降に tty デバイスから読み込まれたキャラクタ数。	キャラクタ
41	vmTotalWriteChars	ブート以降に tty デバイスに書き込まれたキャラクタ数。	キャラクタ
42	vmTotalForks	ブート以降の fork の回数。	-
43	vmTotalForkPages	ブート以降に fork されたページ数。	ページ
44	vmTotalDiskBlockReads	ブート以降に発行されたディスクブロック読み出しの回数。	-
45	vmTotalDiskBlockWrites	ブート以降に発行されたディスクブロック書き込みの回数。	-
46	vmTotalProcessSwapOuts	システムが実行できるプロセスをスワップアウトした回数。この値は 1 秒ごとに更新される。	-
47	vmTotalSwapOutProcesses	システムがスワップアウトした実行できるプロセス数。この値は 1 秒ごとに更新される。	-

#### (4) Process グループ

Process グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-32 Process グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.process )  
( 1.116.5.1.2.1.3 )

ID	オブジェクト名	内容	単位
1	processNum	実行中のプロセス数。	-
2	processTable	実行中のプロセス情報テーブル。	-
2.1	processEntry	各エントリーは、processID の値で識別される。	-
2.1.1	processID	プロセス ID。	-
2.1.2	processIndex	プロセスのエントリー番号。	-

ID	オブジェクト名	内容	単位
2.1.3	processStatus	プロセスの実行状態。 sleep ( 1 ), run ( 2 ), stop ( 3 ), zombie ( 4 ), other ( 5 ), idle ( 6 )	-
2.1.4	processStateFlags	プロセスフラグ。 incore ( 1 ), sys ( 2 ), locked ( 4 ), trace ( 8 ), trace2 ( 16 )	-
2.1.5	processUserID	プロセスの実ユーザー ID。	-
2.1.6	processSavedUserID	プロセスのセーブされたユーザー ID。	-
2.1.7	processParentID	親プロセス ID。	-
2.1.8	processGroupID	プロセスグループ ID。	-
2.1.9	processCPUUtilization	プロセスの CPU 利用率。	-
2.1.10	processPriority	プロセスの優先度。	-
2.1.11	processCPUNice	プロセスのナイス値。	-
2.1.12	processProcessor	プロセスを最後に実行したプロセッサ。	-
2.1.13	processStartTime	プロセスが開始してからの経過時間。	1/100 秒
2.1.14	processPhysicalMemory TextSize	プロセスのテキスト領域物理メモリーサイズ。	バイト
2.1.15	processPhysicalMemory DataSize	プロセスのデータ領域物理メモリーサイズ。	バイト
2.1.16	processPhysicalMemory StackSize	プロセスのスタック領域物理メモリーサイズ。	バイト
2.1.17	processPhysicalMemory SharedMemorySize	プロセスの共用メモリー領域物理メモリーサイズ。	バイト
2.1.18	processPhysicalMemory MemoryMappedSize	プロセスのメモリーマップドファイル領域物理メモリーサイズ。	バイト
2.1.19	processPhysicalMemory UserSize	プロセスのプロセスデータ ( u エリア ) 領域物理メモリーサイズ。	バイト
2.1.20	processPhysicalMemory IOSize	プロセスの I/O 空間物理メモリーサイズ。	バイト
2.1.21	processVirtualMemory TextSize	プロセスのテキスト領域仮想メモリーサイズ。	バイト
2.1.22	processVirtualMemory DataSize	プロセスのデータ領域仮想メモリーサイズ。	バイト
2.1.23	processVirtualMemory StackSize	プロセスのスタック領域仮想メモリーサイズ。	バイト
2.1.24	processVirtualMemory SharedMemorySize	プロセスの共用メモリー領域仮想メモリーサイズ。	バイト
2.1.25	processVirtualMemory MemoryMappedSize	プロセスのメモリーマップドファイル領域仮想メモリーサイズ。	バイト

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.26	processVirtualMemoryUserSize	プロセスのプロセスデータ (u エリア) 領域仮想メモリーサイズ。	バイト
2.1.27	processVirtualMemoryIOSize	プロセスの I/O 空間仮想メモリーサイズ。	バイト
2.1.28	processResidentSize	プロセスの使用メモリーのうちで、ディスクにページアウトされないでメモリー上に残っているメモリーサイズ。	バイト
2.1.29	processAddress	プロセスのユーザー領域アドレス。値は、プロセスがメモリー上にロードされている場合はメモリー中の物理アドレス、スワップアウトされている場合はディスク上のアドレスを示す。	-
2.1.30	processSleepAddress	プロセスがスリープしているアドレス。スリープしていなければ 0。	-
2.1.31	processUserTime	プロセスがユーザーモードで実行した経過時間。	1/100 秒
2.1.32	processSystemTime	プロセスがシステム (カーネル) モードで実行した経過時間。	1/100 秒
2.1.33	processTTYMajor	プロセスが制御端末を持っている場合、tty デバイスマジャー番号を示す。メジャー、マイナーが共に -1 なら、プロセスは制御端末を持たない。	-
2.1.34	processTTYMinor	プロセスが制御端末を持っている場合、tty デバイスマイナー番号を示す。メジャー、マイナーが共に -1 なら、プロセスは制御端末を持たない。	-
2.1.35	processCommand	プロセスを実行したコマンドライン文字列。	-
2.1.36	processExecutable	プロセスの実行ファイル名称文字列。	-
2.1.37	processResidentTime	プロセスのインコアでの経過時間。	秒
2.1.38	processCPUTimeTicks	現在のタイムスライス中にプロセスが消費した CPU のティック数。	-
2.1.39	processTotalCPUTimeTicks	プロセス生成以降、プロセスが消費した CPU ティック数。	-
2.1.40	processFssID	プロセスが所属する均等割り当てスケジューラグループの ID。	-
2.1.41	processResidentTimeCPU	インコアでの経過時間に占める CPU 使用時間の割合。	%
2.1.42	processMinorFaults	プロセスによるページリクレーム数。	-
2.1.43	processMajorFaults	プロセスによって発生したディスクへのアクセスが必要なページフォルト数。	-
2.1.44	processSwapOuts	プロセスがスワップアウトされた回数。	-
2.1.45	processSignals	プロセスが受信したシグナル数。	-
2.1.46	processReceivedMessages	プロセスが受信したメッセージ数。	-
2.1.47	processSentMessages	プロセスが送信したメッセージ数。	-

ID	オブジェクト名	内容	単位
2.1.48	processMaxResidentSize	プロセスが取得できる最大のインコアメモリーサイズ。	バイト
2.1.49	processUser	プロセスを起動したユーザー。	-

### (5) Processor グループ

Processor グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-33 Processor グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.pocessor )  
( 1.116.5.1.2.1.4 )

ID	オブジェクト名	内容	単位
1	processorNum	プロセッサ情報のエントリー数。	-
2	processorTable	プロセッサ情報テーブル。	-
2.1	processorEntry	各エントリーはプロセッサの processorIndex の値で識別される。	-
2.1.1	processorIndex	プロセッサのインデックス番号。	-
2.1.2	processorFileSystemReadBytes	ブート以降にファイルシステムから読み込んだ総バイト数。	バイト
2.1.3	processorFileSystemWriteBytes	ブート以降にファイルシステムに書き込んだ総バイト数。	バイト
2.1.4	processorDiskBlockReadRequests	ブート以降に NFS マウントされたディスクブロックに対して発行した read 要求の総数。	-
2.1.5	processorDiskBlockWriteRequests	ブート以降に NFS マウントされたディスクブロックに対して発行した write 要求の総数。	-
2.1.6	processorNFSReadBytes	ブート以降に NFS から読み込んだ総バイト数。	バイト
2.1.7	processorNFSWriteBytes	ブート以降に NFS に書き込んだ総バイト数。	バイト
2.1.8	processorPhysicalReads	ブート以降にプロセッサによる row デバイスに対する物理読み出しの総数。	-
2.1.9	processorPhysicalWrites	ブート以降にプロセッサによる row デバイスに対する物理書き込みの総数。	-
2.1.10	processorRunQueues	ブート以降にプロセッサ上に実行待ちプロセスが存在した回数。この値は 1 秒ごとに更新される。	-
2.1.11	processorRunQueueProcesses	ブート以降にプロセッサ上の実行待ちプロセス数。この値は 1 秒ごとに更新される。	-
2.1.12	processorSysExecs	ブート以降にプロセッサ上で発行された exec システムコールの総数。	-
2.1.13	processorSysReads	ブート以降にプロセッサ上で発行された read システムコールの総数。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.14	processorSysWrites	ブート以降にプロセッサ上で発行された write システムコールの総数。	-
2.1.15	processorSysNamis	ブート以降にプロセッサ上で発行された sysnami ( ) 関数の総数。	-
2.1.16	processorSysIgets	ブート以降にプロセッサ上で発行された sysiget ( ) 関数の総数。	-
2.1.17	processorDirFileSystemReadBytes	ブート以降にプロセッサ上で行われたディレクトリ参照を伴うファイルシステムリードの総バイト数。	バイト
2.1.18	processorSemaphoreOperations	ブート以降にプロセッサ上で行われた system V 系のセマフォ操作の回数。	-
2.1.19	processorMessageOperations	ブート以降にプロセッサ上で行われた system V 系のメッセージ操作の回数。	-
2.1.20	processorInMUXInterruptions	ブート以降にプロセッサが受け取った MUX 割り込みの回数。	-
2.1.21	processorOutMUXInterruptions	ブート以降にプロセッサが送信した MUX 割り込みの回数。	-
2.1.22	processorTTYRawChars	ブート以降にプロセッサによって行われた raw デバイス読み取りのキャラクタ数。	キャラクタ
2.1.23	processorTTYCanonChars	ブート以降にプロセッサによって行われた canon 操作のキャラクタ数。	キャラクタ
2.1.24	processorTTYOutChars	ブート以降にプロセッサによって行われた、出力キャラクタ数。	キャラクタ
2.1.25	processorCPULoadAvg1	このプロセッサでの過去 1 分間のロードアベレージ * 100。 ロードアベレージとはこのプロセッサで、過去 1 分間に「実行状態」または「実行可能状態」であったプロセス数やスレッド数の平均値のことです。 例えば、ロードアベレージが「1」であった場合、過去 1 分間で「実行状態」または「実行可能状態」となっていたプロセス数やスレッド数が平均的に一つ存在していて、常にこのプロセッサは処理を実行していた状況であったと推測できます。	-
2.1.26	processorCPULoadAvg5	このプロセッサでの過去 5 分間のロードアベレージ * 100。	-
2.1.27	processorCPULoadAvg15	このプロセッサでの過去 15 分間のロードアベレージ * 100。	-
2.1.28	processorUserCPUTime	プロセッサでの、システムがブートしてからのユーザープロセスの CPU 時間。ただし、Solaris, AIX では SNMP エージェントが起動してからのユーザープロセスの CPU 時間。	1/100 秒

ID	オブジェクト名	内容	単位
2.1.29	processorNiceCPUTime	プロセッサでの、システムがブートしてからの nice 値 21 以上で実行されたユーザープロセスの CPU 時間。ただし、Solaris、AIX では SNMP エージェントが起動してからの nice 値が 21 以上で実行されたユーザープロセスの CPU 時間。	1/100 秒
2.1.30	processorSysCPUTime	プロセッサでの、システムがブートしてからのシステム（カーネル）モードで実行されたユーザープロセスの CPU 時間。ただし、Solaris、AIX では SNMP エージェントが起動してからのシステム（カーネル）モードで実行されたユーザープロセスの CPU 時間。	1/100 秒
2.1.31	processorIdleCPUTime	プロセッサでの、システムがブートしてからの CPU アイドル時間。ただし、Solaris、AIX では SNMP エージェントが起動してからの CPU アイドル時間。	1/100 秒
2.1.32	processorWaitCPUTime	プロセッサでの、システムがブートしてからの CPU ウェイト時間。ただし、Solaris、AIX では SNMP エージェントが起動してからの CPU ウェイト時間。	1/100 秒
2.1.33	processorBlockCPUTime	プロセッサでの、システムがブートしてからのスピンロックブロック CPU 時間。	1/100 秒
2.1.34	processorSwaitCPUTime	プロセッサでの、システムがブートしてからのカーネルセマフォブロック CPU 時間。	1/100 秒
2.1.35	processorIntrCPUTime	プロセッサでの、システムがブートしてからの CPU インタラプト時間。	1/100 秒
2.1.36	processorSsysCPUTime	プロセッサでの、システムがブートしてからのカーネルモードで実行されたカーネルプロセスの CPU 時間。	1/100 秒

### (6) DiskInfo グループ

DiskInfo グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-34 DiskInfo グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskInfo )  
( 1.116.5.1.2.1.5 )

ID	オブジェクト名	内容	単位
1	diskNum	システムにアタッチされているディスク情報エントリー数。	-
2	diskTable	ディスク情報テーブル。	-
2.1	diskEntry	各エントリーは、diskIndex の値で識別される。	-
2.1.1	diskIndex	ディスクのインデックス番号。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.2	diskTTYMajor	ディスクの TTY デバイスメジャー番号。	-
2.1.3	diskTTYMinor	ディスクの TTY デバイスマイナー番号。	-
2.1.4	diskBusyTimeTicks	デバイスビジーの TimeTicks 数。	-
2.1.5	diskSeeks	シーク回数。	-
2.1.6	diskXfers	データ転送回数。	-
2.1.7	diskWordsXfers	2 バイト 1 ワードデータの転送回数。	-
2.1.8	diskWordsWriteTime	2 バイト 1 ワードデータの書き込みに要する時間。	ミリ秒 / バイト

#### (7) SwapInfo グループ

SwapInfo グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-35 SwapInfo グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapInfo )  
( 1.116.5.1.2.1.6 )

ID	オブジェクト名	内容	単位
1	swapTotalSize	スワップ領域の全サイズ。	バイト
2	swapTotalEnabledSize	スワップ領域の使用できる全サイズ。	バイト
3	swapTotalFreeSize	スワップ領域の全空きサイズ。	バイト
4	swapTotalBlockDevice Size	ブロックデバイス上に確保されたスワップ領域の全サイズ。	バイト
5	swapTotalBlockDevice EnabledSize	ブロックデバイス上に確保されたスワップ領域の使用できる全サイズ。	バイト
6	swapTotalBlockDevice FreeSize	ブロックデバイス上に確保されたスワップ領域の全空きサイズ。	バイト
7	swapTotalFileSystem Size	ファイルシステム上に確保されたスワップ領域の全サイズ。	バイト
8	swapTotalFileSystem EnabledSize	ファイルシステム上に確保されたスワップ領域の使用できる全サイズ。	バイト
9	swapTotalFileSystem FreeSize	ファイルシステム上に確保されたスワップ領域の全空きサイズ。	バイト
10	swapNum	スワップ領域情報のエントリー数。	-
11	swapTable	スワップ領域情報テーブル。	-
11.1	swapEntry	各エントリーは swapIndex の値で識別される。	-
11.1.1	swapIndex	スワップ領域のインデックス番号。	-

ID	オブジェクト名	内容	単位
11.1.2	swapPlace	スワップ領域がどこに確保されているかを示すフラグ。スワップ領域がブロックデバイス上にある場合は swblock ( 1 ), ファイルシステム上にある場合は swfs ( 2 ) を値として持つ。	-
11.1.3	swapFlags	スワップ領域が有効かどうかを示す。有効の場合 enable ( 1 ), 無効の場合 disable ( 0 ) を値として持つ。	-
11.1.4	swapPriority	スワップ領域の優先順位。値の小さいスワップ領域から先に使用される。	-
11.1.5	swapFreeSize	現在使用できるスワップ領域中のフリーサイズ。	バイト
11.1.6	swapBlockDeviceMajor	スワップ領域がブロックデバイス上に確保された場合に、デバイス情報 ( Major ) が格納される。	-
11.1.7	swapBlockDeviceMinor	スワップ領域がブロックデバイス上に確保された場合に、デバイス情報 ( Minor ) が格納される。	-
11.1.8	swapBlockDeviceStart Num	スワップ領域がブロックデバイス上に確保された場合に、使用を開始するブロック番号が格納される。	-
11.1.9	swapBlockDeviceSize	スワップ領域がブロックデバイス上に確保された場合の、確保されたサイズ。	バイト
11.1.10	swapFileSystemSize	スワップ領域がファイルシステム上に確保された場合の、確保されたサイズ。	バイト
11.1.11	swapFileSystemMinSize	スワップ領域がファイルシステム上に確保された場合の、最低限割り当てられるサイズ。	バイト
11.1.12	swapFileSystemMax Size	スワップ領域がファイルシステム上に確保された場合に、今までに割り当てられた最大のサイズ。	バイト
11.1.13	swapFileSystem ReservedSize	スワップ領域がファイルシステム上に確保された場合の、現在リザーブ中のサイズ。	バイト
11.1.14	swapFileSystemMount Point	スワップ領域がファイルシステム上に確保された場合の、ファイルシステムがマウントされている場所。	-

### ( 8 ) SwapSpace グループ

SwapSpace グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-36 SwapSpace グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSpace )  
( 1.116.5.1.2.1.7 )

ID	オブジェクト名	内容	単位
1	swapSpaceConfig	スワップ空間のサイズ。	キロバイト
2	swapSpaceEnable	使用できるスワップ空間のサイズ。	キロバイト
3	swapSpaceFree	空きスワップ空間のサイズ。	キロバイト

#### 4. MIB オブジェクト一覧

### (9) DiskBusy グループ

DiskBusy グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-37 DiskBusy グループ

(enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusy)  
(1.116.5.1.2.1.8)

ID	オブジェクト名	内容	単位
1	diskBusyNum	diskBusyTable のエントリー数。	-
2	diskBusyTable	ディスクビジー率を表すテーブル。	-
2.1	diskBusyEntry	各エントリーは diskBusyDeviceName で識別される。	-
2.1.1	diskBusyDeviceName	ディスクデバイス名。 SNMP エージェントを Solaris にインストールしている場合、ディスク単位にエントリーを作成する。ディスク内の個々のパーティション、またはスライスと呼ばれる区画のエントリーは作成しない。 例えば、dad0 や sd3 などが表示される。 ここでは、“dad” は内蔵 IDE ディスク，“sd” は SCSI ディスクを表している。そのあとの数字は、ターゲット ID を表している。	-
2.1.2	diskBusyUtil	過去 5 秒間のディスクビジー率。 <ul style="list-style-type: none"> <li>ディスクビジー率 カーネルは、アプリケーションプログラムから出された I/O 要求をデバイスドライバに送る。デバイスドライバから、デバイスに対して I/O 要求を出す。この要求は、デバイス内に持っているキューバッファにためられる。このキューの要求でディスクがシークし、実際に読み込みをし、逆の経路をたどって戻る。ディスクビジー率は、このディスクそのものが処理している時間の割合を示す。 このディスクビジー率は、収集した時点のディスクビジー率である。このディスクビジー率が複数回同じ状態が継続するなどの状況からディスクの状態を判断できる。</li> <li>取得時の注意事項 SNMP エージェントを Solaris にインストールしている場合、diskBusyTable のインスタンスを取得するには 6 秒以上掛かる。そのため、マネージャシステムの SNMP リクエストのタイムアウト時間には、6 秒以上を指定する。</li> </ul>	%

### (10) SwapSystem グループ

SwapSystem グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-38 SwapSystem グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSystem )  
 ( 1.116.5.1.2.1.9 )

ID	オブジェクト名	内容	単位
1	swapSystemTotalSize	スワップ領域の全サイズ。	キロバイト
2	swapSystemTotalEnableSize	スワップ領域の使用できる全サイズ。	キロバイト
3	swapSystemTotalFreeSize	スワップ領域の全空きサイズ。	キロバイト
4	swapSystemTotalBlockDeviceSize	ブロックデバイス上に確保されたスワップ領域の全サイズ。	キロバイト
5	swapSystemTotalBlockDeviceEnabledSize	ブロックデバイス上に確保されたスワップ領域の使用できる全サイズ。	キロバイト
6	swapSystemTotalBlockDeviceFreeSize	ブロックデバイス上に確保されたスワップ領域の全空きサイズ。	キロバイト
7	swapSystemTotalFileSystemSize	ファイルシステム上に確保されたスワップ領域の全サイズ。	キロバイト
8	swapSystemTotalFileSystemEnabledSize	ファイルシステム上に確保されたスワップ領域の使用できる全サイズ。	キロバイト
9	swapSystemTotalFileSystemFreeSize	ブロックデバイス上に確保されたスワップ領域の全空きサイズ。	キロバイト
10	swapSystemNum	スワップ領域情報のエントリー数。	-
11	swapSystemTable	スワップ領域情報テーブル。	-
11.1	swapSystemEntry	各エントリーは swapSystemIndex の値で識別される。	-
11.1.1	swapSystemIndex	スワップ領域のインデックス番号。	-
11.1.2	swapSystemPlace	スワップ領域がどこに確保されているかを示すフラグ。 swblock(1) : ブロックデバイス上にある。 swfs(2) : ファイルシステム上にある。	-
11.1.3	swapSystemFlags	スワップ領域が有効かどうかを示す。 disable(0) : 無効 enable(1) : 有効	-
11.1.4	swapSystemPriority	スワップ領域の優先順位。値の小さいスワップ領域から先に使用される。	-
11.1.5	swapSystemFreeSize	現在使用できるスワップ領域中のフリーサイズ。	キロバイト
11.1.6	swapSystemBlockDeviceMajor	スワップ領域がブロックデバイス上に確保された場合に、デバイス情報 (Major) が格納される。	-
11.1.7	swapSystemBlockDeviceMinor	スワップ領域がブロックデバイス上に確保された場合に、デバイス情報 (Minor) が格納される。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
11.1.8	swapSystemBlockDeviceStartNum	スワップ領域がブロックデバイス上に確保された場合に、使用を開始するブロック番号が格納される。	-
11.1.9	swapSystemBlockDeviceSize	スワップ領域がブロックデバイス上に確保された場合に、確保されたサイズ。	キロバイト
11.1.10	swapSystemFileSystemSize	スワップ領域がファイルシステム上に確保された場合に、確保されたサイズ。	キロバイト
11.1.11	swapSystemFileSystemMinSize	スワップ領域がファイルシステム上に確保された場合に、今まで割り当てられた最大のサイズ。	キロバイト
11.1.12	swapSystemFileSystemMaxSize	スワップ領域がファイルシステム上に確保された場合に、今までに割り当てられた最大のサイズ。	キロバイト
11.1.13	swapSystemFileSystemReservedSize	スワップ領域がファイルシステム上に確保された場合に、リザーブ中のサイズ。	キロバイト
11.1.14	swapSystemFileSystemMountPoint	スワップ領域がファイルシステム上に確保された場合に、ファイルシステムがマウントされている場所を示す。	-

#### ( 11 )cpuUtil グループ

cpuUtil グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-39 cpuUtil グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.cpuUtil )  
( 1.116.5.1.2.1.10 )

ID	オブジェクト名	内容	単位
1	cpuUtilTable	CPU 利用率情報。	-
1.1	cpuUtilEntry	各エントリーは cpuUtilNum の値で識別される。SNMP エージェントは指定されたインターバル時間 ( デフォルトは 5 分 ) の CPU 利用率を OS から取得して MIB 値とする。また、この CPU 利用率の取得を連続して実行する。マネージャーからこの MIB を定期的に収集することで個々の CPU 利用率を監視することができる。	-
1.1.1	cpuUtilNum	CPU 番号 ( CPU が 1 個の場合は 0 )	-
1.1.2	cpuUtilUser	指定されたインターバル時間内のユーザー CPU 利用率	%
1.1.3	cpuUtilSystem	指定されたインターバル時間内のシステム CPU 利用率	%
1.1.4	cpuUtilWio	指定されたインターバル時間内のウェイト CPU 利用率	%
1.1.5	cpuUtilIdle	指定されたインターバル時間内のアイドル CPU 利用率	%
1.1.6	cpuUtilTime	CPU 利用率を OS から取得した時刻 例 : 2003/01/16 19:00:00	-

ID	オブジェクト名	内容	単位
2	cpuUtilInterval	インターバル時間	分
3	cpuUtilTotalUser	すべての CPU の指定されたインターバル時間内のユーザー CPU 利用率の平均値	%
4	cpuUtilTotalSystem	すべての CPU の指定されたインターバル時間内のシステム CPU 利用率の平均値	%
5	cpuUtilTotalWio	すべての CPU の指定されたインターバル時間内のウェイト CPU 利用率の平均値	%
6	cpuUtilTotalIdle	すべての CPU の指定されたインターバル時間内のアイドル CPU 利用率の平均値	%

### 注意事項

cpuUtil グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- インターバル時間（分）は htc\_monagt1 のオプションで変更できます。
- SNMP エージェントが MIB 値を取得しないように設定することができます。
- Solaris で CPU の online/offline 状態を変更した場合、OS から CPU 利用率の情報が取得できないため、一時的に上記の表の ID 2 以外の MIB 値はすべて noSuchName エラーで応答します。その後、CPU 利用率取得のインターバル時間が経過して OS から CPU 利用率の情報が取得できるようになれば、すべての MIB 値は取得できるようになります。また、CPU の online/offline 状態が変更されても取得する CPU 番号には変更はありません。
- AIX で CPU がダイナミックに追加 / 削除（DLPAR：Dynamic Logical Partition）される場合があります。CPU の追加 / 削除を実行した場合、OS から CPU 利用率の情報が取得できないため一時的に上記の表の ID 2 以外の MIB 値はすべて noSuchName エラーで応答します。その後 CPU 利用率取得のインターバル時間が経過して OS から CPU 利用率の情報が取得できるようになれば、すべての MIB 値はできるようになります。
- cpuUtilTotalUser, cpuUtilTotalSystem, cpuUtilTotalWio, cpuUtilTotalIdle は個々の CPU の利用率を加算し CPU 数で割り算した結果の小数点以下を切り捨てた値のため、合計値が 100% とならないことがあります。  
SMT 環境では、SNMP\_HTC\_AIX\_CPU\_SMT 環境変数でマシン全体の CPU 利用率を取得する指定ができます。ただし、JP1/Cm2/SSO からリソースブラウザで CPU 利用率を取得したとき、実際に複数の CPU を搭載した環境であっても、マシン全体で一つのインスタンスとして CPU 情報が表示されます。

### ( 12 ) virtualMemory64 グループ

virtualMemory64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

#### 4. MIB オブジェクト一覧

表 4-40 virtualMemory64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory64 )  
( 1.116.5.1.2.1.13 )

ID	オブジェクト名	内容	単位
1	vm64PageSize	仮想記憶システムでの 1 ページのバイト数。	バイト
2	vm64DaemonfreePages	ページアウトデーモンにフリーされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
3	vm64Interruptions	デバイスからの割り込み回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
4	vm64PageInPages	ページインされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
5	vm64PageOutPages	ページアウトされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
6	vm64PageReclaims	ページリクレームの合計数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
7	vm64TLBFlashes	TLB フラッシュの回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
8	vm64DaemonScanPages	ページアウトデーモンにスキャンされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
9	vm64ContextSwitches	コンテキストスイッチの回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
10	vm64SystemCalls	システムコールの発行回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
11	vm64XFileSystemFreelistPages	ファイルシステムからではなく、ページフリーリストから見つかったページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
12	vm64XSwapDeviceFreeListPages	スワップデバイスからではなく、ページフリーリストから見つかったページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
13	vm64FreeMemoryPages	フリーメモリーページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
14	vm64TotalSwapIns	ブート以降のスワップインの回数。	-
15	vm64TotalSwapOuts	ブート以降のスワップアウトの回数。	-
16	vm64TotalDaemonFreePages	ブート以降に、ページアウトデーモンにフリーされたページ数。	ページ
17	vm64TotalDemandLoadPages	ブート以降に、実行可能ファイルにデマンドロードされたページ数。	ページ
18	vm64TotalPageFaults	ブート以降のページフォルト回数。	-

ID	オブジェクト名	内容	単位
19	vm64TotalInterruptions	ブート以降のデバイス割り込み回数。	-
20	vm64TotalIntransitPageFaults	ブート以降の intransit ブロックページフォルトの回数。	-
21	vm64TotalDemandLoadCreatePages	ブート以降に、デマンドロードページで新たに作成されたページ数。	ページ
22	vm64TotalZeroFillCreatePages	ブート以降に、ゼロフィルページで新たに作成されたページ数。	ページ
23	vm64TotalFreeListReclaimedPages	ブート以降に、フリーリストからリクレーンされたページ数。	ページ
24	vm64TotalPageIns	ブート以降のページイン回数。	-
25	vm64TotalPageOuts	ブート以降のページアウト回数。	-
26	vm64TotalPageInPages	ブート以降にページインされたページ数。	ページ
27	vm64TotalPageOutPages	ブート以降にページアウトされたページ数。	ページ
28	vm64TotalSwapInPages	ブート以降にスワップインされたページ数。	ページ
29	vm64TotalSwapOutPages	ブート以降にスワップアウトされたページ数。	ページ
30	vm64TotalDaemonTicksNum	ブート以降にページアウトデーモンがページアウトした回数。	-
31	vm64TotalContextSwitches	ブート以降のコンテキストスイッチの回数。	-
32	vm64TotalSystemCalls	ブート以降に発行されたシステムコールの回数。	-
33	vm64TotalTraps	ブート以降に発生したトラップの回数。	-
34	vm64TotalXFileSystemFreeListPages	ブート以降に、ファイルシステムからではなくフリーリストから見つかったページ数。	ページ
35	vm64TotalXSwapDeviceFreeListPages	ブート以降に、スワップデバイスからではなくフリーリストから見つかったページ数。	ページ
36	vm64TotalDemandZeroFillPages	ブート以降にオンデマンドでゼロフィルされたページ数。	ページ
37	vm64TotalDaemonScanPages	ブート以降にページアウトデーモンにスキャンされたページ数。	ページ
38	vm64TotalReclaimedPages	ブート以降にリクレーンされたページ数。	ページ
39	vm64TotalDeficitPages	新たにスワップインされようとしているプロセスに必要と思われるページ数。	ページ
40	vm64TotalReadChars	ブート以降に tty デバイスから読み込まれたキャラクタ数。	キャラクタ
41	vm64TotalWriteChars	ブート以降に tty デバイスに書き込まれたキャラクタ数。	キャラクタ
42	vm64TotalForks	ブート以降の fork 回数。	-
43	vm64TotalForkPages	ブート以降に fork されたページ数。	ページ

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
44	vm64TotalDiskBlockReads	ブート以降に発行されたディスクブロックリードの回数。	-
45	vm64TotalDiskBlockWrites	ブート以降に発行されたディスクブロックライトの回数。	-
46	vm64TotalProcessSwapOuts	システムが実行可能なプロセスをスワップアウトした回数。この値は1秒ごとに更新される。	-
47	vm64TotalSwapOutProcesses	システムがスワップアウトした実行可能なプロセス数。この値は1秒ごとに更新される。	-

#### 注意事項

virtualMemory64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 14 ~ 47 のオブジェクトの SYNTAX は Counter64 です。SYNTAX が Counter64 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。

#### ( 13 )systemInfo64 グループ

systemInfo64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-41 systemInfo64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.systemInfo64 )  
( 1.116.5.1.2.1.14 )

ID	オブジェクト名	内容	単位
1	systemInfo64TTYMajor	制御端末のメジャー番号。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
2	systemInfo64TTYMinor	制御端末のマイナー番号。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
3	systemInfo64BootTime	システムがブートしてからの経過時間。	1/100 秒
4	systemInfo64ActiveProcessors	システムコールを発行した時点で動作しているプロセッサ数。	-
5	systemInfo64MaxProcessors	システムがブートされてから今までに同時に動作したことがある最大のプロセッサの数。	-
6	systemInfo64MaxProcesses	システムが動作させることのできる最大のプロセス数。	-
7	systemInfo64RunQueProcesses	システムプロセス以外の実行待ち状態のプロセスで、主記憶装置上にロードされているプロセスの数。	-

ID	オブジェクト名	内容	単位
8	systemInfo64XferWaitProcesses	ディスクからのデータ転送を持っているプロセス数。	-
9	systemInfo64PageInWaitProcesses	ディスクからページインされるのを待っているプロセスの数。	-
10	systemInfo64SleepProcesses	物理メモリー中でスリープしているプロセス数。ただし、次の状態のプロセスは含まない。 <ul style="list-style-type: none"> <li>• I/O 待ちでスリープしているプロセス</li> <li>• 連続して 20 秒以上スリープしているプロセス</li> </ul>	-
11	systemInfo64SwapOutProcesses	スワップアウトされているプロセスの数。ただし、スワップインされたらすぐに実行待ち状態になるプロセスだけである。	-
12	systemInfo64PhysicalMemorySize	システムに搭載されている物理メモリーの最大のサイズ。	バイト
13	systemInfo64PhysicalMemoryFreeSize	未使用で、かつシステムがプロセスに割り当てできるとみなした物理メモリーのサイズ。	バイト
14	systemInfo64VirtualMemoryProcessSize	システムプロセス以外のすべてのテキスト、データ、スタックとして割り当てられている仮想メモリーのサイズ。	バイト
15	systemInfo64VirtualMemoryWaitProcessSize	実行待ち状態のすべてのプロセスに割り当てられている仮想メモリーのサイズ。	バイト
16	systemInfo64PhysicalMemoryProcessSize	システム内のすべてのプロセスの、テキスト、データ、スタックに割り当てられている物理メモリーのサイズ。	バイト
17	systemInfo64PhysicalMemoryWaitProcessSize	実行待ち状態のすべてのプロセスに割り当てられている物理メモリーのサイズ。	バイト
18	systemInfo64CPUStates	システムが持つ CPU 状態の数。	-
19	systemInfo64OpenLogicalVolumes	オープン済みの論理ボリュームの数。	-
20	systemInfo64OpenLogicalVolumeGrps	オープン済みの論理ボリュームグループの数。	-
21	systemInfo64AllocPBUFs	論理ボリューム用に割り当てられている pbuf の数。	-
22	systemInfo64UsedPBUFs	論理ボリュームで使用している pbuf の数。	-
23	systemInfo64MaxPBUFs	今まで論理ボリュームによって使用された pbuf の最大数。	-
24	systemInfo64ActiveProcessEntries	プロセステーブルの総エントリー数のうちで現在使用中のエントリー数。	エントリー
25	systemInfo64ActiveInodeEntries	i ノードテーブルの総エントリー数のうちで現在使用中のエントリー数。	エントリー
26	systemInfo64ActiveFileEntries	ファイルテーブルの総エントリー数のうちで現在使用中のエントリー数。	エントリー

#### 4. MIB オブジェクト一覧

##### 注意事項

systemInfo64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 3 ~ 26 のオブジェクトの SYNTAX は CounterBasedGauge64 です。SYNTAX が CounterBasedGauge64 のオブジェクトは SNMPv2c リクエストでだけ取得できません。SNMPv1 リクエストでは取得できません。

##### ( 14 ) virtualMemory64Ex グループ

virtualMemory64Ex グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-42 virtualMemory64Ex グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory64Ex )  
( 1.116.5.1.2.1.15 )

ID	オブジェクト名	内容	単位
1	vm64ExPageSize	仮想記憶システムでの 1 ページのバイト数。	バイト
2	vm64ExDaemonfreePages	ページアウトデーモンにフリーされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
3	vm64ExInterruptions	デバイスからの割り込み回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
4	vm64ExPageInPages	ページインされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
5	vm64ExPageOutPages	ページアウトされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
6	vm64ExPageReclaims	ページリクレームの合計数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
7	vm64ExTLBFlashes	TLB フラッシュの回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
8	vm64ExDaemonScanPages	ページアウトデーモンにスキャンされたページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ
9	vm64ExContextSwitches	コンテキストスイッチの回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
10	vm64ExSystemCalls	システムコールの発行回数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	-
11	vm64ExXFileSystemFreelistPages	ファイルシステムからではなく、ページフリーリストから見つかったページ数。この値は 1 秒当たりの平均値で、5 秒ごとに更新される。	ページ

ID	オブジェクト名	内容	単位
12	vm64ExXSwapDeviceFreeListPages	スワップデバイスからではなく、ページフリーリストから見つかったページ数。この値は1秒当たりの平均値で、5秒ごとに更新される。	ページ
13	vm64ExFreeMemoryPages	フリーメモリーページ数。この値は1秒当たりの平均値で、5秒ごとに更新される。	ページ
14	vm64ExTotalSwapIns	ブート以降のスワップインの回数。	-
15	vm64ExTotalSwapOuts	ブート以降のスワップアウトの回数。	-
16	vm64ExTotalDaemonFreePages	ブート以降に、ページアウトデーモンにフリーされたページ数。	ページ
17	vm64ExTotalDemandLoadPages	ブート以降に、実行可能ファイルにデマンドロードされたページ数。	ページ
18	vm64ExTotalPageFaults	ブート以降のページフォルト回数。	-
19	vm64ExTotalInterruptions	ブート以降のデバイス割り込み回数。	-
20	vm64ExTotalIntransitPageFaults	ブート以降の intransit ブロックページフォルトの回数。	-
21	vm64ExTotalDemandLoadCreatePages	ブート以降に、デマンドロードページで新たに作成されたページ数。	ページ
22	vm64ExTotalZeroFillCreatePages	ブート以降に、ゼロフィルページで新たに作成されたページ数。	ページ
23	vm64ExTotalFreeListReclaimedPages	ブート以降に、フリーリストからリクレームされたページ数。	ページ
24	vm64ExTotalPageIns	ブート以降のページイン回数。	-
25	vm64ExTotalPageOuts	ブート以降のページアウト回数。	-
26	vm64ExTotalPageInPages	ブート以降にページインされたページ数。	ページ
27	vm64ExTotalPageOutPages	ブート以降にページアウトされたページ数。	ページ
28	vm64ExTotalSwapInPages	ブート以降にスワップインされたページ数。	ページ
29	vm64ExTotalSwapOutPages	ブート以降にスワップアウトされたページ数。	ページ
30	vm64ExTotalDaemonTicksNum	ブート以降にページアウトデーモンがページアウトした回数。	-
31	vm64ExTotalContextSwitches	ブート以降のコンテキストスイッチの回数。	-
32	vm64ExTotalSystemCalls	ブート以降に発行されたシステムコールの回数。	-
33	vm64ExTotalTraps	ブート以降に発生したトラップの回数。	-
34	vm64ExTotalXFileSystemFreeListPages	ブート以降に、ファイルシステムからではなくフリーリストから見つかったページ数。	ページ
35	vm64ExTotalXSwapDeviceFreeListPages	ブート以降に、スワップデバイスからではなくフリーリストから見つかったページ数。	ページ
36	vm64ExTotalDemandZeroFillPages	ブート以降にオンデマンドでゼロフィルされたページ数。	ページ

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
37	vm64ExTotalDaemonScanPages	ブート以降にページアウトデーモンにスキャンされたページ数。	ページ
38	vm64ExTotalReclaimedPages	ブート以降にリクレーンされたページ数。	ページ
39	vm64ExTotalDeficitPages	新たにスワップインされようとしているプロセスに必要と思われるページ数。	ページ
40	vm64ExTotalReadChars	ブート以降に tty デバイスから読み込まれたキャラクタ数。	キャラクタ
41	vm64ExTotalWriteChars	ブート以降に tty デバイスに書き込まれたキャラクタ数。	キャラクタ
42	vm64ExTotalForks	ブート以降の fork 回数。	-
43	vm64ExTotalForkPages	ブート以降に fork されたページ数。	ページ
44	vm64ExTotalDiskBlockReads	ブート以降に発行されたディスクブロックリードの回数。	-
45	vm64ExTotalDiskBlockWrites	ブート以降に発行されたディスクブロックライトの回数。	-
46	vm64ExTotalProcessSwapOuts	システムが実行可能なプロセスをスワップアウトした回数。この値は 1 秒ごとに更新される。	-
47	vm64ExTotalSwapOutProcesses	システムがスワップアウトした実行可能なプロセス数。この値は 1 秒ごとに更新される。	-

#### 注意事項

virtualMemory64Ex グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 1 ~ 13 のオブジェクトの SYNTAX は CounterBasedGauge64 です。ID が 14 ~ 47 のオブジェクトの SYNTAX は Counter64 です。SYNTAX が CounterBasedGauge64 または Counter64 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。

#### ( 15 ) process64 グループ

process64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-43 process64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.process64 )  
( 1.116.5.1.2.1.16 )

ID	オブジェクト名	内容	単位
1	process64Num	実行中のプロセス数。	-
2	process64Table	実行中のプロセス情報テーブル。	-

ID	オブジェクト名	内容	単位
2.1	process64Entry	各エントリはプロセス ID で識別される。	-
2.1.1	process64ID	プロセス ID。	-
2.1.2	process64Index	プロセスのインデックス番号。	-
2.1.3	process64Status	プロセスの実行状態。sleep ( 1 ), run ( 2 ), stop ( 3 ), zombie ( 4 ), other ( 5 ), idle ( 6 )	-
2.1.4	process64StatusFlags	プロセス状態フラグ。incore ( 1 ), sys ( 2 ), locked ( 4 ), trace ( 8 ), trace2 ( 16 )	-
2.1.5	process64UserID	プロセスの実ユーザー ID。	-
2.1.6	process64SavedUserID	プロセスのセーブされたユーザー ID。	-
2.1.7	process64ParentID	親プロセス ID。	-
2.1.8	process64GroupID	プロセスグループ ID。	-
2.1.9	process64CPUUtilization	プロセスの CPU 利用率。	-
2.1.10	process64Priority	プロセスの優先度。	-
2.1.11	process64CPUNice	プロセスのナイス値。	-
2.1.12	process64Processor	プロセスを最新に実行したプロセッサ。	-
2.1.13	process64StartTime	プロセスが開始してからの経過時間。	1/100 秒
2.1.14	process64PhysicalMemoryTextSize	テキストとして割当てられた物理メモリのサイズ。	バイト
2.1.15	process64PhysicalMemoryDataSize	データとして割当てられた物理メモリのサイズ。	バイト
2.1.16	process64PhysicalMemoryStackSize	スタックとして割当てられた物理メモリのサイズ。	バイト
2.1.17	process64PhysicalMemorySharedMemorySize	共有メモリとして割当てられた物理メモリのサイズ。	バイト
2.1.18	process64PhysicalMemoryMemoryMappedSize	メモリーマップドファイルとして割当てられた物理メモリのサイズ。	バイト
2.1.19	process64PhysicalMemoryUserSize	プロセスデータ ( u エリア ) として割当てられた物理メモリのサイズ。	バイト
2.1.20	process64PhysicalMemoryIOSize	I/O 空間として割当てられた物理メモリのサイズ。	バイト
2.1.21	process64VirtualMemoryTextSize	テキストとして割当てられた仮想アドレス空間のサイズ。	バイト
2.1.22	process64VirtualMemoryDataSize	データとして割当てられた仮想アドレス空間のサイズ。	バイト
2.1.23	process64VirtualMemoryStackSize	スタックとして割当てられた仮想アドレス空間のサイズ。	バイト
2.1.24	process64VirtualMemorySharedMemorySize	共有メモリとして割当てられた仮想アドレス空間のサイズ。	バイト

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.25	process64VirtualMemoryMemoryMappedSize	メモリーマップドファイルとして割当てられた仮想アドレス空間のサイズ。	バイト
2.1.26	process64VirtualMemoryUserSize	プロセスデータ (u エリア) として割当てられた仮想アドレス空間のサイズ。	バイト
2.1.27	process64VirtualMemoryIOSize	I/O 空間として割当てられた仮想アドレス空間のサイズ。	バイト
2.1.28	process64ResidentSize	プロセスの使用メモリーの内、ディスクにページアウトされないで、メモリー上に残っているメモリーサイズ。	バイト
2.1.29	process64Address	プロセスのユーザー領域アドレス。その値は、プロセスがメモリー上にロードされている場合はメモリー中の物理アドレス、スワップアウトされている場合はディスク上のアドレスを示す。	-
2.1.30	process64SleepAddress	プロセスが休眠しているアドレス。休眠していなければ 0。	-
2.1.31	process64UserTime	プロセスがユーザーモードで実行した経過時間。	1/100 秒
2.1.32	process64SystemTime	プロセスがシステムモードで実行した経過時間。	1/100 秒
2.1.33	process64TTYMajor	プロセスが制御端末を持っている場合、tty デバイスメジャー番号を示す。メジャー、マイナー共に - 1 なら、このプロセスは制御端末を持たない。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
2.1.34	process64TTYMinor	プロセスが制御端末を持っている場合、tty デバイスマイナー番号を示す。メジャー、マイナー共に - 1 なら、このプロセスは制御端末を持たない。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
2.1.35	process64Command	コマンドライン文字列。	-
2.1.36	process64Executable	実行ファイル名称文字列。	-
2.1.37	process64ResidentTime	インコアでの経過時間。	秒
2.1.38	process64CPUTimeTicks	現在のタイムスライス中にプロセスが消費した CPU のティック数。	-
2.1.39	process64TotalCPUTimeTicks	プロセス生成以降、プロセスが消費した CPU ティック数。	-
2.1.40	process64FssID	プロセスが所属する均等割り当てスケジューラグループの ID。	-
2.1.41	process64ResidentTimeCPU	インコアでの経過時間に占める CPU 使用時間の割合。	%
2.1.42	process64MinorFaults	このプロセスによって行われたページリクレーム数。	-

ID	オブジェクト名	内容	単位
2.1.43	process64MajorFaults	このプロセスによって発生したディスクアクセスが必要なページフォルト数。	-
2.1.44	process64SwapOuts	このプロセスがスワップアウトされた回数。	-
2.1.45	process64Signals	このプロセスが受信したシグナル数。	-
2.1.46	process64ReceivedMessages	このプロセスが受信したメッセージ数。	-
2.1.47	process64SentMessages	このプロセスが送信したメッセージ数。	-
2.1.48	process64MaxResidentSize	このプロセスが取得できる最大のインコアメモリーサイズ。	バイト
2.1.49	process64User	このプロセスを起動したユーザー。	-

### 注意事項

process64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 2.1.2 , 2.1.5 ~ 2.1.7 , 2.1.9 , 2.1.11 ~ 2.1.32 , 2.1.37 , 2.1.48 のオブジェクトの SYNTAX は CounterBasedGauge64 です。ID が 2.1.38 ~ 2.1.39 , 2.1.42 ~ 2.1.47 のオブジェクトの SYNTAX は Counter64 です。SYNTAX が CounterBasedGauge64 または Counter64 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。

### ( 16 )processor64 グループ

processor64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-44 processor64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.processor64 )  
( 1.116.5.1.2.1.17 )

ID	オブジェクト名	内容	単位
1	processor64Num	プロセッサ情報のエントリー数。	-
2	processor64Table	プロセッサ情報テーブル。	-
2.1	processor64Entry	各エントリーはプロセッサの processor64Index の値で識別される。	-
2.1.1	processor64Index	プロセッサのインデックス番号。	-
2.1.2	processor64FileSystemReadBytes	ブート以降にファイルシステムから読み込んだ総バイト数。	バイト
2.1.3	processor64FileSystemWriteBytes	ブート以降にファイルシステムから書き込んだ総バイト数。	バイト
2.1.4	processor64DiskBlockReadRequests	ブート以降に NFS マウントされたディスクブロックに対して発行した read 要求の総数。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.5	processor64DiskBlockWriteRequests	ブート以降に NFS マウントされたディスクブロックに対して発行した write 要求の総数。	-
2.1.6	processor64NFSReadBytes	ブート以降に NFS から読み込んだ総バイト数。	バイト
2.1.7	processor64NFSWriteBytes	ブート以降に NFS から書き込んだ総バイト数。	バイト
2.1.8	processor64PhysicalReads	ブート以降にプロセッサによる row デバイスに対する物理読み出しの総数。	-
2.1.9	processor64PhysicalWrites	ブート以降にプロセッサによる row デバイスに対する物理書き込みの総数。	-
2.1.10	processor64RunQueues	ブート以降にプロセッサ上に実行待ちプロセスがあった回数。この値は 1 秒ごとに更新される。	-
2.1.11	processor64RunQueueProcesses	ブート以降にプロセッサ上に実行待ちプロセス数。この値は 1 秒ごとに更新される。	-
2.1.12	processor64SysExecs	ブート以降にプロセッサ上で発行された exec システムコールの総数。	-
2.1.13	processor64SysReads	ブート以降にプロセッサ上で発行された read システムコールの総数。	-
2.1.14	processor64SysWrites	ブート以降にプロセッサ上で発行された write システムコールの総数。	-
2.1.15	processor64SysNamis	ブート以降にプロセッサ上で発行された sysnami ( ) 関数の総数。	-
2.1.16	processor64SysIgets	ブート以降にプロセッサ上で発行された sysiget ( ) 関数の総数。	-
2.1.17	processor64DirFileSystemReadBytes	ブート以降にプロセッサ上で行われたディレクトリ参照を伴うファイルシステムリードの総バイト数。	バイト
2.1.18	processor64SemaphoreOperations	ブート以降にプロセッサ上で行われた systemV 系のセマフォ操作の回数。	-
2.1.19	processor64MessageOperations	ブート以降にプロセッサ上で行われた systemV 系のメッセージ操作の回数。	-
2.1.20	processor64InMUXInterruptions	ブート以降にプロセッサが受け取った MUX 割り込みの回数。	-
2.1.21	processor64OutMUXInterruptions	ブート以降にプロセッサが送信した MUX 割り込みの回数。	-
2.1.22	processor64TTYRawChars	ブート以降にプロセッサによって行われた raw デバイス読み取りのキャラクタ数。	キャラクタ
2.1.23	processor64TTYCanonChars	ブート以降にプロセッサによって行われた canon 操作のキャラクタ数。	キャラクタ

ID	オブジェクト名	内容	単位
2.1.24	processor64TTYOutChars	ブート以降にプロセッサによって行われた、出力キャラクタ数。	キャラクタ
2.1.25	processor64CPULoadAvg1	このプロセッサでの過去 1 分間のロードアベレージ * 100。 ロードアベレージとはこのプロセッサで過去 1 分間に「実行状態」または「実行可能状態」であったプロセス数やスレッド数の平均値のことである。 例えば、ロードアベレージが「1」であった場合、過去 1 分間で「実行状態」または「実行可能状態」となっていたプロセス数やスレッド数が平均的に一つ存在していて、常にこのプロセッサは処理を実行していた状況であったと推測できる。	-
2.1.26	processor64CPULoadAvg5	このプロセッサでの過去 5 分間のロードアベレージ * 100。	-
2.1.27	processor64CPULoadAvg15	このプロセッサでの過去 15 分間のロードアベレージ * 100。	-
2.1.28	processor64UserCPUTime	このプロセッサでの、システムがブートしてからのユーザープロセスの CPU 時間。	1/100 秒
2.1.29	processor64NiceCPUTime	このプロセッサでの、システムがブートしてからの nice 値 21 以上で実行されたユーザープロセスの CPU 時間。	1/100 秒
2.1.30	processor64SysCPUTime	このプロセッサでの、システムがブートしてからのシステム（カーネル）モードで実行されたユーザープロセスの CPU 時間。	1/100 秒
2.1.31	processor64IdleCPUTime	このプロセッサでの、システムがブートしてからの CPU アイドル時間。	1/100 秒
2.1.32	processor64WaitCPUTime	このプロセッサでの、システムがブートしてからのウェイト時間。	1/100 秒
2.1.33	processor64BlockCPUTime	このプロセッサでの、システムがブートしてからのスピンロックブロック CPU 時間。	1/100 秒
2.1.34	processor64SwaitCPUTime	このプロセッサでの、システムがブートしてからのカーネルセマフォブロック CPU 時間。	1/100 秒
2.1.35	processor64IntrCPUTime	このプロセッサでの、システムがブートしてからの CPU インタラプト時間。	1/100 秒
2.1.36	processor64SsysCPUTime	このプロセッサでの、システムがブートしてからのカーネルモードで実行されたカーネルプロセスの CPU 時間。	1/100 秒

#### 注意事項

processor64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 2.1.2 ~ 2.1.24, 2.1.28 ~ 2.1.36 のオブジェクトの SYNTAX は Counter64 で

#### 4. MIB オブジェクト一覧

す。SYNTAX が Counter64 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。

#### ( 17 )diskInfo64 グループ

diskInfo64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-45 diskInfo64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskInfo64 )  
( 1.116.5.1.2.1.18 )

ID	オブジェクト名	内容	単位
1	disk64Num	システムにアタッチされているディスク情報エントリー数。	-
2	disk64Table	ディスク情報テーブル。	-
2.1	disk64Entry	各エントリーは、disk64Index の値で識別される。	-
2.1.1	disk64Index	ディスクのインデックス番号。	-
2.1.2	disk64TTYMajor	ディスクの TTY デバイスメジャー番号。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
2.1.3	disk64TTYMinor	ディスクの TTY デバイスマイナー番号。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
2.1.4	disk64BusyTimeTicks	デバイスビジーの TimeTicks 数。	-
2.1.5	disk64Seeks	シーク数。	-
2.1.6	disk64Xfers	データ転送回数。	-
2.1.7	disk64WordsXfers	2 バイト 1 ワードデータの転送回数。	-
2.1.8	disk64WordsWriteTime	2 バイト 1 ワードデータの書き込みに要する時間。	ミリ秒 / バイト

#### 注意事項

diskInfo64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 2.1.4 ~ 2.1.7 のオブジェクトの SYNTAX は Counter64 です。SYNTAX が Counter64 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。

#### ( 18 )swapSystem64 グループ

swapSystem64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-46 swapSystem64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSystem64 )  
 ( 1.116.5.1.2.1.19 )

ID	オブジェクト名	内容	単位
1	swapSystem64TotalSize	スワップ領域の全サイズ。	キロバイト
2	swapSystem64TotalEnabledSize	スワップ領域の使用できる全サイズ。	キロバイト
3	swapSystem64TotalFreeSize	スワップ領域の全空きサイズ。	キロバイト
4	swapSystem64TotalBlockDeviceSize	ブロックデバイス上に確保されたスワップ領域の全サイズ。	キロバイト
5	swapSystem64TotalBlockDeviceEnabledSize	ブロックデバイス上に確保されたスワップ領域の使用できる全サイズ。	キロバイト
6	swapSystem64TotalBlockDeviceFreeSize	ブロックデバイス上に確保されたスワップ領域の全空きサイズ。	キロバイト
7	swapSystem64TotalFileSystemSize	ファイルシステム上に確保されたスワップ領域の全サイズ。	キロバイト
8	swapSystem64TotalFileSystemEnabledSize	ファイルシステム上に確保されたスワップ領域の使用できる全サイズ。	キロバイト
9	swapSystem64TotalFileSystemFreeSize	ブロックデバイス上に確保されたスワップ領域の全空きサイズ。	キロバイト
10	swapSystem64Num	スワップ領域情報のエントリー数。	-
11	swapSystem64Table	スワップ領域情報テーブル。	-
11.1	swapSystem64Entry	各エントリーは swapSystem64Index の値で識別される。	-
11.1.1	swapSystem64Index	スワップ領域のインデックス番号。	-
11.1.2	swapSystem64Place	スワップ領域がどこに確保されているかを示すフラグ。 swblock ( 1 ): ブロックデバイス上にある swfs ( 2 ): ファイルシステム上にある	-
11.1.3	swapSystem64Flags	スワップ領域が有効かどうかを示す。 disable ( 0 ): 無効 enable ( 1 ): 有効	-
11.1.4	swapSystem64Priority	スワップ領域の優先順位。値の小さいスワップ領域から先に使用される。	-
11.1.5	swapSystem64FreeSize	現在使用できるスワップ領域中のフリーサイズ。	キロバイト
11.1.6	swapSystem64BlockDeviceMajor	スワップ領域がブロックデバイス上に確保された場合、デバイス情報 ( Major ) が格納される。値が ( $2^{31} - 1$ ) 以上なら、( $2^{31} - 1$ ) にする。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
11.1.7	swapSystem64BlockDeviceMinor	スワップ領域がブロックデバイス上に確保された場合、デバイス情報 (Minor) が格納される。値が $(2^{31} - 1)$ 以上なら、 $(2^{31} - 1)$ にする。	-
11.1.8	swapSystem64BlockDeviceStart Num	スワップ領域がブロックデバイス上に確保された場合、使用を開始するブロック番号が格納される。	-
11.1.9	swapSystem64BlockDeviceSize	スワップ領域がブロックデバイス上に確保された場合の、確保されたサイズ。	キロバイト
11.1.10	swapSystem64FileSystemSize	スワップ領域がファイルシステム上に確保された場合の、確保されたサイズ。	キロバイト
11.1.11	swapSystem64FileSystemMinSize	スワップ領域がファイルシステム上に確保された場合の、今まで割り当てられた最大のサイズ。	キロバイト
11.1.12	swapSystem64FileSystemMaxSize	スワップ領域がファイルシステム上に確保された場合の、今までに割り当てられた最大のサイズ。	キロバイト
11.1.13	swapSystem64FileSystemReservedSize	スワップ領域がファイルシステム上に確保された場合の、リザーブ中のサイズ。	キロバイト
11.1.14	swapSystem64FileSystemMount Point	スワップ領域がファイルシステム上に確保された場合の、ファイルシステムがマウントされている場所を示す。	-

#### 注意事項

swapSystem64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- ID が 1 ~ 9 , 11.1.4 ~ 1.1.5 , 11.1.8 ~ 11.1.13 のオブジェクトの SYNTAX は CounterBasedGauge64 です。SYNTAX が CounterBasedGauge64 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。

#### ( 19 ) diskBusyInfo グループ

diskBusyInfo グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-47 diskBusyInfo グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusyInfo )  
( 1.116.5.1.2.1.20 )

ID	オブジェクト名	内容	単位
1	diskBusyInfoTable	ディスクビジー時間を表すテーブル。	-

ID	オブジェクト名	内容	単位
1.1	diskBusyInfoEntry	各エントリは diskBusyInfoDeviceName で識別される。	-
1.1.1	diskBusyInfoDeviceName	ディスクデバイス名。 ディスク単位にエントリを作成する。ディスク内の個々のパーティションまたはスライスと呼ばれる区画のエントリは作成しない。	-
1.1.2	diskBusyInfoDeviceIndex	SNMP エージェントが付加するインデックス値で、OS からの取得情報で同一デバイス名があった場合にカウントアップする。	-
1.1.3	diskBusyInfoTime	SNMP エージェント起動時からのディスクビジーの累計時間。	1/100 秒
2	diskBusyInfoNum	diskBusyInfoTable のエントリ数。	-

## ( 20 )fileSystem64 グループ

fileSystem64 グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-48 fileSystem64 グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.fileSystem64 )  
( 1.116.5.1.2.1.21 )

ID	オブジェクト名	内容	単位
1	fileSystem64Mounted	マウントされているファイルシステムの数。	-
2	fileSystem64Table	ファイルシステム情報テーブル。	-
2.1	fileSystem64Entry	各エントリは fileSystem64HighID1 , fileSystem64LowID1 , および fileSystem64ID2 の値で識別される。	-
2.1.1	fileSystem64HighID1	ファイルシステム ID の high bit。	-
2.1.2	fileSystem64LowID1	ファイルシステム ID の low bit。	-
2.1.3	fileSystem64ID2	OS から取得した fileSystem64HighID1 と fileSystem64LowID1 が同一の場合にカウントアップする。	-
2.1.4	fileSystem64Name	マウントされているファイルシステム名。	-
2.1.5	fileSystem64Block	ファイルシステム中のトータルブロック数。	ブロック
2.1.6	fileSystem64Bfree	ファイルシステム中のフリーブロック数。	ブロック
2.1.7	fileSystem64Bavail	スーパーユーザーではない場合の利用できるフリーブロック数。	ブロック
2.1.8	fileSystem64Bsize	基本ファイルシステムのブロックサイズ。	バイト
2.1.9	fileSystem64Files	ファイルシステム中の i ノード総数。	-
2.1.10	fileSystem64Ffree	ファイルシステム中の未使用 i ノード総数。	-

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.11	fileSystem64Dir	ファイルシステムの path。	-

##### 注意事項

fileSystem64 グループの日立企業固有 MIB オブジェクトの注意事項を次に示します。

- fileSystem64 グループの ID が 2.1.5 ~ 2.1.10 のオブジェクトは SNMPv2c リクエストでだけ取得できます。SNMPv1 リクエストでは取得できません。
- AIX, Linux, および Linux (IPF) の場合, クラスタシステムで論理ホストが存在しないノードから共有ディスクのファイルシステム情報を取得すると, 不正な情報を応答する場合があります。これは対象ファイルシステムがマウントされていないためです。SNMP エージェントが対象となるファイルシステムがマウントされているか, マウントされていないかをチェックして, マウントされていない場合には対象ファイルシステムの情報を応答しないようにすることができます。

次の作業をスーパーユーザーで実施します。

1. /etc/SnmpAgent.d/esafilesys.conf ファイルをエディタで開きます。
2. /etc/SnmpAgent.d/esafilesys.conf ファイルの最後に次の行を追加します。

check: 共有ディスクのファイルシステムpath名 共有ディスク直下に存在する任意のファイル名

(例)

共有ディスク /shdisk1, /shdisk2 を監視する。

test1, test2 ファイルはそれぞれ /shdisk1, /shdisk2 直下に存在する。

check: /shdisk1 test1

check: /shdisk2 test2

#### 3. SNMP エージェントを再起動します。

SNMP エージェントは起動時に /etc/SnmpAgent.d/esafilesys.conf を読み込みます。その際, /etc/SnmpAgent.d/esafilesys.conf ファイルの内容にシンタクスエラーがあった場合はその行を無視して起動します。シンタクスエラーの内容は /etc/SnmpAgent.d/esafilesys.conf.err に出力します。

check: ラベルの指定方法を次に示します。

- check: と共有ディスクのファイルシステム path, および共有ディスクのファイルシステム path と共有ディスク直下に存在する任意のファイル名の間には必ず 1 文字の空白を挿入してください。
- 共有ディスクのファイルシステム path または共有ディスク直下に存在する任意のファイル名の最大長は 1024 文字です。
- 複数の共有ディスクを指定する場合は, 行を追加して定義してください。
- クラスタシステムのローカルノードにある共有ディスクのファイルシステム path ディレクトリに, /etc/SnmpAgent.d/esafilesys.conf ファイルに記述した共有ディス

ク直下に存在する任意のファイル名と同じ名前のファイルが存在しないことを必ず確認してください。

- 共有ディスクのファイルシステム path は次の内容を指定してください。

Linux, Linux (IPF) の場合: /etc/fstab ファイルの fs\_file フィールド

AIX の場合: /etc/filesystems の 1 番目フィールド ( : を除く )

- Linux, Linux (IPF) の場合, SNMP エージェントが cdrom または floppy がマウントされていないのに, cdrom または floppy の情報を応答します。これは OS の情報からだけではマウントされているのかマウントされていないのが認識できないためです。このような応答不要なファイルシステムが存在する場合は次の方法を実施することで SNMP エージェントが対象となるファイルシステムの情報を応答しないようにすることができます。

次の作業をスーパーユーザーで実施します。

1. /etc/SnmpAgent.d/esafilesys.conf ファイルをエディタで開きます。
2. /etc/SnmpAgent.d/esafilesys.conf ファイルの最後に次の行を追加します。

```
exclude: SNMPエージェントが応答しないファイルシステムのpath名
```

exclude: とファイルシステムの path 名の間には必ず 1 文字の空白を挿入してください。

path 名の最大長は 1024 文字です。

(例)

/mnt/cdrom, /mnt/floppy には応答しないようにする場合

```
exclude: /mnt/cdrom
```

```
exclude: /mnt/floppy
```

3. SNMP エージェントを再起動します。

SNMP エージェントは起動時に /etc/SnmpAgent.d/esafilesys.conf を読み込みます。その際、/etc/SnmpAgent.d/esafilesys.conf ファイルの内容にシンタクスエラーがあった場合はその行を無視して起動します。シンタクスエラーの内容は /etc/SnmpAgent.d/esafilesys.conf.err に出力します。

- AIX の場合, fileSystem64Block の最大値は  $2^{32}-1$  個です。そのため, fileSystem64Bsize が 1024 バイトの場合, ファイルシステム容量の最大値は 4 テラバイトです。また, ファイルシステム中の i ノード総数の最大値は  $2^{32}-1$  個です。
- Linux, Linux (IPF) の場合, fileSystem64Block の最大値は  $2^{32}-1$  個です。そのため, fileSystem64Bsize が 1024 バイトの場合, ファイルシステム容量の最大値は 4 テラバイトです。また, ファイルシステム中の i ノード総数の最大値は  $2^{32}-1$  個です。ただし, 2.4.21-27.EL カーネル以降では, ファイルシステム容量が 4 テラバイトを超える場合やファイルシステム中の i ノード総数が  $2^{32}-1$  個を超える場合でも fileSystem64Block の MIB オブジェクトを取得できます。そのためには, スーパーユーザーで次に示す手順を実行する必要があります。

#### 4. MIB オブジェクト一覧

- 次に示すファイルを vi などのエディタを使用して編集します。

```
/opt/CM2/ESA/opt/SnmpHtcunix1
```

- 次の行をファイルの最終行に追加します。

```
SNMP_HTC_FILE_EXTEND=1
```

```
export SNMP_HTC_FILE_EXTEND
```

- ファイルを保存します。

- fileSystem64 グループで監視できるファイルシステムのタイプを次に示します。なお、NFS を監視する場合、NFS サーバ側でファイルシステムを監視するようにしてください。

##### HP-UX (IPF) の場合

/etc/mnttab に記述されているファイルシステム。

ただし、タイプが swap , ignore , nfs のファイルシステムは監視できません。

##### Solaris の場合

/etc/mnttab に記述されているファイルシステム。

ただし、タイプが nfs , swap のファイルシステムは監視できません。

##### AIX の場合

/etc/filesystems に記述されているファイルシステム。

##### Linux および Linux (IPF) の場合

/etc/fstab に記述されているファイルシステム。

ファイルタイプが swap の場合は /etc/fstab に記述されていても応答しません。

これは、SNMP エージェントがファイルシステムを取得するために発行しているシステムコールがこのファイルタイプを応答しないためです。

### ( 21 ) diskBusyAvail グループ

diskBusyAvail グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-49 diskBusyAvail グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusyAvail )  
( 1.116.5.1.2.1.22 )

ID	オブジェクト名	内容	単位
1	diskBusyAvailTable	ディスクビジー利用率を表すテーブル。	-
1.1	diskBusyAvailEntry	各エントリは diskBusyAvailDeviceName, diskBusyAvailDeviceIndex で識別される。	-
1.1.1	diskBusyAvailDeviceName	ディスクデバイス名。ディスク単位にエントリを作成する。ディスク内の個々のパーティションまたはスライスと呼ばれる区画のエントリは作成しない。	-
1.1.2	diskBusyAvailDeviceIndex	SNMP エージェントが付加するインデックス値で、OS からの取得情報で同一デバイス名があった場合にカウントアップする。	-

ID	オブジェクト名	内容	単位
1.1.3	diskBusyAvailDiskBusy	指定されたインターバル時間内のディスクビジー利用率。	%
1.1.4	diskBusyAvailTime	ディスクビジー利用率を OS から取得した時刻。 例：2004/11/11 14:47:00	-
2	diskBusyAvailNum	DiskBusyAvail のエントリー数。	-
3	diskBusyAvailInterval	インターバル時間。	分

注 インターバル時間（分）は htc\_monagt1 のオプションで変更できます。

## (22) system グループ (AIX の場合)

system グループ (AIX の場合) の日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-50 system グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.aix.system )  
( 1.116.5.1.2.3.1 )

ID	オブジェクト名	内容	単位
1	systemCPUutilUser	ユーザー CPU 利用率 (過去 1 秒間の平均値)。	%
2	systemCPUutilSystem	システム CPU 利用率 (過去 1 秒間の平均値)。	%
3	systemCPUutilIdle	アイドル CPU 利用率 (過去 1 秒間の平均値)。	%
4	systemCPUutilWait	ウェイト CPU 利用率 (過去 1 秒間の平均値)。	%
5	systemActVirtualPage	アクティブな仮想ページ数。	ページ

## (23) disk グループ (AIX の場合)

disk グループ (AIX の場合) の日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-51 disk グループ ( enterprises.hitachi.systemExMib.CometMibs.systems.aix.disk )  
( 1.116.5.1.2.3.2 )

ID	オブジェクト名	内容	単位
1	diskNum	ディスク情報のエントリー数。	-
2	diskTable	ディスクの入出力情報。	-
2.1	diskTableEntry	各エントリーは diskIndex の番号で識別される。	-
2.1.1	diskIndex	インデックス情報。	-
2.1.2	diskName	ディスク名称。	-
2.1.3	diskRead	このディスクが読み込んだデータ量。	キロバイト

#### 4. MIB オブジェクト一覧

ID	オブジェクト名	内容	単位
2.1.4	diskWrite	このディスクが書き込んだデータ量。	キロバイト

#### (24) page グループ (AIX の場合)

page グループ (AIX の場合) の日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-52 page グループ (enterprises.hitachi.systemExMib.CometMibs.systems.aix.page) (1.116.5.1.2.3.3)

ID	オブジェクト名	内容	単位
1	pageNum	ページングスペースのエントリー数。	-
2	pageTable	ページングスペースの構成および使用率。	-
2.1	pageTableEntry	各エントリーは、pageIndex の番号で識別される。	-
2.1.1	pageIndex	インデックス番号。	-
2.1.2	pageSpaceName	ページングスペースの名称。	-
2.1.3	pagePhysicalVol	ページングスペースの物理ボリューム名。	-
2.1.4	pageVolGroup	ページングスペースのボリュームグループ名。	-
2.1.5	pageSize	ページングスペースのサイズ。	メガバイト
2.1.6	pageUsed	ページングスペースの使用率。	%
2.1.7	pageActive	ページングスペースが現在使用されているかどうか。 1: inactive (使用されていない) 2: active (使用されている)	-
2.1.8	pageAuto	ページングスペースが OS の起動時から使用されるかどうか。 1: not auto (使用されない) 2: auto (使用される)	-
2.1.9	pageType	ページングスペースのタイプ。 1: lv (論理ボリューム) 2: nfs (NFS ボリューム)	-

#### (25) system グループ (Solaris の場合)

system グループ (Solaris の場合) の日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-53 system グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.solaris.system )  
 ( 1.116.5.1.2.4.1 )

ID	オブジェクト名	内容	単位
1	systemCPUUtilUser	ユーザー CPU 利用率 (過去 1 秒間の平均値)	%
2	systemCPUUtilSystem	システム CPU 利用率 (過去 1 秒間の平均値)	%
3	systemCPUUtilIdle	アイドル CPU 利用率 (過去 1 秒間の平均値)	%
4	systemCPUUtilWait	ウェイト CPU 利用率 (過去 1 秒間の平均値)	%

### ( 26 )LinuxSystem グループ ( Linux の場合 )

LinuxSystem グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-54 LinuxSystem グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.Linux.linuxsystem )  
 ( 1.116.5.1.2.5.1 )

ID	オブジェクト名	内容	単位
1	linuxSystemCPUUtilUser	ユーザー CPU 利用率 (過去 1 秒間の平均値)	%
2	linuxSystemCPUUtilSystem	システム CPU 利用率 (過去 1 秒間の平均値)	%
3	linuxSystemCPUUtilIdle	アイドル CPU 利用率 (過去 1 秒間の平均値)	%

### ( 27 )Tru64system グループ ( Tru64 の場合 )

Tru64system グループの日立企業固有 MIB オブジェクトの内容を次の表に示します。

表 4-55 Tru64system グループ

( enterprises.hitachi.systemExMib.CometMibs.systems.Tru64.Tru64system )  
 ( 1.116.5.1.2.6.1 )

ID	オブジェクト名	内容	単位
1	tru64SystemCPUUtilUser	ユーザー CPU 利用率 (過去 1 秒間の平均値)	%
2	tru64SystemCPUUtilSystem	システム CPU 利用率 (過去 1 秒間の平均値)	%
3	tru64SystemCPUUtilIdle	アイドル CPU 利用率 (過去 1 秒間の平均値)	%

## 4.3.3 日立企業固有 MIB オブジェクトの実装状況

SNMP エージェントが動作するシステムによって、日立企業固有 MIB オブジェクトのグループが異なります。日立企業固有 MIB オブジェクトのグループの適用 OS と参照先

#### 4. MIB オブジェクト一覧

を次の表に示します。

表 4-56 日立企業固有 MIB オブジェクトの実装状況の適用 OS と参照先

グループ名 (上位ノード)	適用 OS				MIB オブジェクトの実装状況の参照先
	Solaris	AIX	Linux, Linux (IPF)	HP-UX (IPF)	
OpConf ( cometOP )					表 4-57
SystemInfo ( hiux )	-		-	-	表 4-58
VirtualMemory ( hiux )	-		-	-	表 4-59
Process ( hiux )	-		-	-	表 4-60
Processor ( hiux )				-	表 4-61
DiskInfo ( hiux )	-	-	-	-	表 4-62
SwapInfo ( hiux )	-	-	-	-	表 4-63
SwapSpace ( hiux )	-		-	-	表 4-64
DiskBusy ( hiux )		-	-	-	表 4-65
SwapSystem ( hiux )	-	-	-	-	表 4-66
cpuUtil ( hiux )				-	表 4-67
virtualMemory64 ( hiux )	-		-	-	表 4-68
systemInfo64 ( hiux )	-	-	-		表 4-69
virtualMemory64Ex ( hiux )	-	-	-		表 4-70
process64 ( hiux )	-	-	-		表 4-71
processor64 ( hiux )	-	-	-		表 4-72
diskInfo64 ( hiux )	-	-	-		表 4-73
swapSystem64 ( hiux )	-	-	-		表 4-74
diskBusyInfo ( hiux )			-	-	表 4-75
fileSystem64 ( hiux )					表 4-76
diskBusyAvail ( hiux )			-	-	表 4-77
system ( aix )	-		-	-	表 4-78
disk ( aix )	-		-	-	
page ( aix )	-		-	-	
system ( solaris )		-	-	-	表 4-79
LinuxSystem ( Linux )	-	-		-	表 4-80
Tru64system ( Tru64 )	-	-	-	-	表 4-81

( 凡例 ) : 適用する。 - : 適用しない。

注 SwapSystem (hiux) および Tru64System グループ (Tru64) は日立企業固有 MIB オブジェクトの MIB オペレーションには実装されていません。

注 Linux (IPF) の場合は適用しません。

グループごとの日立企業固有 MIB オブジェクトの実装状況を、表 4-57 ~ 表 4-81 に示します。なお、表中では次の凡例を使用しています。

(凡例)

: get オペレーションで値を取得できる。または set オペレーションで値をセットできる。

N : get オペレーションで値を取得できない。または set オペレーションで値をセットできない。noSuchName で応答する。

F (値):(値) で示す固定値で応答する。

- : アクセス権限がない。noSuchName で応答する。

### (1) OpConf グループ

OpConf グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-57 日立企業固有 MIB オブジェクトの実装状況 (OpConf グループ)

(enterprises.hitachi.systemExMib.CometMibs.SubSystems.cometOP.OpConf)  
(1.116.5.1.1.2.10)

オブジェクト名	MIB オペレーション							
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux , Linux (IPF) の場合	
	get	set	get	set	get	set	get	set
opConfCharCode		-		-		-		-

### (2) SystemInfo グループ

SNMP エージェントは AIX の SystemInfo グループをサポートしています。

SystemInfo グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

4. MIB オブジェクト一覧

表 4-58 日立企業固有 MIB オブジェクトの実装状況 ( SystemInfo グループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.systemInfo )  
 ( 1.116.5.1.2.1.1 )

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
systemInfoTTYMajor	N	-	-	-
systemInfoTTYMinor	N	-	-	-
systemInfoBootTime	N	-	-	-
systemInfoActiveProcessors	N	-	-	-
systemInfoMaxProcessors	N	-	-	-
systemInfoMaxProcesses	N	-	-	-
systemInfoRunQue Processes	N	-	-	-
systemInfoXferWait Processes	N	-	-	-
systemInfoPageInWait Processes	N	-	-	-
systemInfoSleepProcesses	N	-	-	-
systemInfoSwapOut Processes	N	-	-	-
systemInfoPhysicalMemorySize	N	-	-	-
systemInfoPhysicalMemoryFreeSize	N	-	-	-
systemInfoVirtualMemory ProcessSize	N	-	-	-
systemInfoVirtualMemory WaitProcessSize	N	-	-	-
systemInfoPhysicalMemoryProcessSize	N	-	-	-
systemInfoPhysicalMemoryWaitProcessSize	N	-	-	-
systemInfoCPUStates	N	-	-	-
systemInfoOpenLogical Volumes	N	-	-	-
systemInfoOpenLogical VolumeGrps	N	-	-	-
systemInfoAllocPBUFs	N	-	-	-
systemInfoUsedPBUFs	N	-	-	-
systemInfoMaxPBUFs	N	-	-	-
systemInfoActiveProcess Entries	N	-	-	-

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
systemInfoActiveInode Entries	N	-	-	-
systemInfoActiveFileEntries	N	-	-	-

### ( 3 ) VirtualMemory グループ

SNMP エージェントは AIX の VirtualMemory グループをサポートしています。

VirtualMemory グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-59 日立企業固有 MIB オブジェクトの実装状況 ( VirtualMemory グループ )  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory )  
( 1.116.5.1.2.1.2 )

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
vmPageSize	N	-	-	-
vmDaemonFreePages	N	-	-	-
vmInterruptions	N	-	-	-
vmPageInPages	N	-	-	-
vmPageOutPages	N	-	-	-
vmPageReclaims	N	-	-	-
vmTLBFlashes	N	-	-	-
vmDaemonScanPages	N	-	-	-
vmContextSwitches	N	-	-	-
vmSystemCalls	N	-	-	-
vmXfileSystemFreeList Pages	N	-	-	-
vmXSwapDeviceFreeList Pages	N	-	-	-
vmFreeMemoryPages	N	-	-	-
vmTotalSwapIns	N	-	-	-
vmTotalSwapOuts	N	-	-	-
vmTotalDaemonFreePages	N	-	-	-

#### 4. MIB オブジェクト一覧

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
vmTotalDemandLoadPages	N	-	-	-
vmTotalPageFaults	N	-	-	-
vmTotalInterruptions	N	-	-	-
vmTotalIntransitPageFaults	N	-	-	-
vmTotalDemandLoadCreatePages	N	-	-	-
vmTotalZeroFillCreatePages	N	-	-	-
vmTotalFreeListReclaimedPages	N	-	-	-
vmTotalPageIns	N	-	-	-
vmTotalPageOuts	N	-	-	-
vmTotalPageInPages	N	-	-	-
vmTotalPageOutPages	N	-	-	-
vmTotalSwapInPages	N	-	-	-
vmTotalSwapOutPages	N	-	-	-
vmTotalDaemonTicksNum	N	-	-	-
vmTotalContextSwitches	N	-	-	-
vmTotalSystemCalls	N	-	-	-
vmTotalTraps	N	-	-	-
vmTotalXfileSystemFreeListPages	N	-	-	-
vmTotalXSwapDeviceFreeListPages	N	-	-	-
vmTotalDemandZeroFillPages	N	-	-	-
vmTotalDaemonScanPages	N	-	-	-
vmTotalReclaimedPages	N	-	-	-
vmTotalDeficitPages	N	-	-	-
vmTotalReadChars	N	-	-	-
vmTotalWriteChars	N	-	-	-
vmTotalForks	N	-	-	-
vmTotalForkPages	N	-	-	-
vmTotalDiskBlockReads	N	-	-	-
vmTotalDiskBlockWrites	N	-	-	-
vmTotalProcessSwapOuts	N	-	-	-

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
vmTotalswapOutProcesses	N	-	-	-

#### (4) Process グループ

SNMP エージェントは AIX の Process グループをサポートしています。

Process グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-60 日立企業固有 MIB オブジェクトの実装状況 (Process グループ)  
 (enterprises.hitachi.systemExMib.CometMibs.systems.hiux.process)  
 (1.116.5.1.2.1.3)

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
processNum	N	-	-	-
process Table. process Entry. ~	processID	N	-	-
	processIndex	N	-	-
	processStatus	N	-	-
	processState Flags	N	-	-
	processUserID	N	-	-
	processSaved UserID	N	-	-
	processParentID	N	-	-
	processGroupID	N	-	-
	processCPU Utilization	N	-	-
	processPriority	N	-	-
	processCPUNice	N	-	-
	processProcessor	N	-	-
	processStartTime	N	-	-
processPhysical MemoryTextSize	N	-	-	

#### 4. MIB オブジェクト一覧

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
processPhysicalMemoryDataSize	N	-	-	-
processPhysicalMemoryStackSize	N	-	-	-
processPhysicalMemorySharedMemorySize	N	-	-	-
processPhysicalMemoryMemoryMappedSize	N	-	-	-
processPhysicalMemoryUserSize	N	-	-	-
processPhysicalMemoryIOSize	N	-	-	-
processVirtualMemoryTextSize	N	-	-	-
processVirtualMemoryDataSize	N	-	-	-
processVirtualMemoryStackSize	N	-	-	-
processVirtualMemorySharedMemorySize	N	-	-	-
processVirtualMemoryMemoryMappedSize	N	-	-	-
processVirtualMemoryUserSize	N	-	-	-
processVirtualMemoryIOSize	N	-	-	-
processResidentSize	N	-	-	-
processAddress	N	-	-	-
processSleepAddress	N	-	-	-
processUserTime	N	-	-	-

オブジェクト名	MIB オペレーション			
	Solaris の場合		AIX の場合	
	get	set	get	set
processSystem Time	N	-	-	-
processTTYMajor	N	-	-	-
processTTYMinor	N	-	-	-
processCommand	N	-	-	-
process Executable	N	-	-	-
processResident Time	N	-	-	-
processCPUTimeTicks	N	-	-	-
processTotalCPUTimeTicks	N	-	-	-
processFssID	N	-	-	-
processResident TimeCPU	N	-	-	-
processMinor Faults	N	-	-	-
processMajor Faults	N	-	-	-
processSwapOuts	N	-	-	-
processSignals	N	-	-	-
processReceived Messages	N	-	-	-
processSent Messages	N	-	-	-
processMax ResidentSize	N	-	-	-
processUser	N	-	-	-

### (5) Processor グループ

SNMP エージェントは HP-UX (IPF) および Linux (IPF) の Processor グループをサポートしていません。

Processor グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

4. MIB オブジェクト一覧

表 4-61 日立企業固有 MIB オブジェクトの実装状況 (Processor グループ)  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.pocessor )  
 ( 1.116.5.1.2.1.4 )

オブジェクト名		MIB オペレーション					
		Solaris の場合		AIX の場合		Linux の場合	
		get	set	get	set	get	set
processorNum			-		-		-
processorTable. processorEntry. ~	processorIndex		-		-		-
	processorFileSystemReadBytes	N	-	-	-	-	-
	processorFileSystemWriteBytes	N	-	-	-	-	-
	processorDiskBlockReadRequests	N	-	-	-	-	-
	processorDiskBlockWriteRequests	N	-	-	-	-	-
	processorNFSReadBytes	N	-	-	-	-	-
	processorNFSWriteBytes	N	-	-	-	-	-
	processorPhysicalReads	N	-	-	-	-	-
	processorPhysicalWrites	N	-	-	-	-	-
	processorRunQueues	N	-	-	-	-	-
	processorRunQueueProcesses	N	-	-	-	-	-
	processorSysExecs	N	-	-	-	-	-
	processorSysReads	N	-	-	-	-	-
	processorSysWrites	N	-	-	-	-	-

オブジェクト名		MIB オペレーション					
		Solaris の場合		AIX の場合		Linux の場合	
		get	set	get	set	get	set
processorSys Namis	N	-	-	-	-	-	-
processor SysIgets	N	-	-	-	-	-	-
processorDir FileSystem ReadBytes	N	-	-	-	-	-	-
processor Semaphore Operations	N	-	-	-	-	-	-
processo rMessage Operations	N	-	-	-	-	-	-
processor InMUX Interruptions	N	-	-	-	-	-	-
processor OutMUX Interruptions	N	-	-	-	-	-	-
processor TTYRawChars	N	-	-	-	-	-	-
processor TTYCanon Chars	N	-	-	-	-	-	-
processor TTYOutChars	N	-	-	-	-	-	-
processorCPU LoadAvg1	N	-	-	-	-	-	-
processorCPU LoadAvg5	N	-	-	-	-	-	-
processorCPU LoadAvg15	N	-	-	-	-	-	-
processorUser CPUTime		-		-			-
processorNice CPUTime	-	-	-	-			-
processorSys CPUTime		-		-			-
processorIdle CPUTime		-		-			-

#### 4. MIB オブジェクト一覧

オブジェクト名		MIB オペレーション					
		Solaris の場合		AIX の場合		Linux の場合	
		get	set	get	set	get	set
	processorWait CPUTime		-		-	-	-
	processorBlockCPUTime	N	-	-	-	-	-
	processorSwaitCPUTime	N	-	-	-	-	-
	processorIntr CPUTime	N	-	-	-	-	-
	processorSsys CPUTime	N	-	-	-	-	-

注

「4.2.2(1) ComputerSystem グループ」の CPU 情報についての注意事項を参照してください。

#### (6) DiskInfo グループ

SNMP エージェントは DiskInfo グループをサポートしていません。

表 4-62 日立企業固有 MIB オブジェクトの実装状況 (DiskInfo グループ)  
(enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskInfo)  
(1.116.5.1.2.1.5)

オブジェクト名		MIB オペレーション	
		get	set
diskNum		実装されていません。	
diskTable. diskEntry ..	diskIndex		
	diskTTYMajor		
	disk TTYMinor		
	diskBusyTime Ticks		
	diskSeeks		
	diskXfers		
	disk WordsXfers		
diskWords WriteTime			

## (7) SwapInfo グループ

SNMP エージェントは SwapInfo グループをサポートしていません。

表 4-63 日立企業固有 MIB オブジェクトの実装状況 (SwapInfo グループ)  
(enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapInfo)  
(1.116.5.1.2.1.6)

オブジェクト名		MIB オペレーション	
		get	set
swapTotalSize		実装されていません。	
swapTotalEnabledSize			
swapTotalFreeSize			
swapTotalBlockDeviceSize			
swapTotalBlockDevice EnabledSize			
swapTotalBlockDevice FreeSize			
swapTotalFreeSystemSize			
swapTotalFileSystem EnabledSize			
swapTotalFileSystem FreeSize			
swapNum			
swapTable. swapEntry. ~	swapIndex		
	swapPlace		
	swapFlags		
	swapPriority		
	swap FreeSize		
	swapBlock DeviceMajor		
	swapBlock DeviceMinor		
	swapBlock DeviceStart Num		
	swapFile SystemSize		

#### 4. MIB オブジェクト一覧

オブジェクト名		MIB オペレーション	
		get	set
	swapFile System MinSize		
	swapFile System MaxSize		
	swapFile System ReservedSize		
	swapFile System MountPoint		

#### (8) SwapSpace グループ

SNMP エージェントは AIX の SwapSpace グループをサポートしています。

SwapSpace グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-64 日立企業固有 MIB オブジェクトの実装状況 (SwapSpace グループ)  
(enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSpace)  
(1.116.5.1.2.1.7)

オブジェクト名	MIB オペレーション					
	Solaris の場合		AIX の場合		Linux の場合	
	get	set	get	set	get	set
swapSpaceConfig	-	-		-	-	-
swapSpaceEnable	-	-		-	-	-
swapSpaceFree	-	-		-	-	-

#### (9) DiskBusy グループ

SNMP エージェントは、Solaris の DiskBusy グループをサポートしています。

DiskBusy グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-65 日立企業固有 MIB オブジェクトの実装状況 (DiskBusy グループ)  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusy )  
 ( 1.116.5.1.2.1.8 )

オブジェクト名		MIB オペレーション	
		Solaris の場合	
		get	set
diskBusyNum			-
diskBusyTable. diskBusyEntry ~	diskBusyDeviceName		-
	diskBusyUtil		-

### ( 10 ) SwapSystem グループ

SNMP エージェントは SwapSystem グループをサポートしていません。

表 4-66 日立企業固有 MIB オブジェクトの実装状況 (SwapSystem グループ)  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSystem )  
 ( 1.116.5.1.2.1.9 )

オブジェクト名		MIB オペレーション	
		get	set
swapSystemTotalSize		実装されていません。	
swapSystemTotalEnableSize			
swapSystemTotalFreeSize			
swapSystemTotalBlockDeviceSize			
swapSystemTotalBlockDeviceEnabledSize			
swapSystemTotalBlockDeviceFreeSize			
swapSystemTotalFileSystemSize			
swapSystemTotalFileSystemEnabledSize			
swapSystemTotalFileSystemFreeSize			
swapSystemNum			
swapSystemTable. swapSystemEntry ~	swapSystemIndex		
	swapSystemPlace		
	swapSystemFlags		
	swapSystemPriority		
	swapSystemFreeSize		

#### 4. MIB オブジェクト一覧

オブジェクト名	MIB オペレーション	
	get	set
swapSystemBlockDeviceMajor		
swapSystemBlockDeviceMinor		
swapSystemBlockDeviceStartNum		
swapSystemBlockDeviceSize		
swapSystemFileSystemSize		
swapSystemFileSystemMinSize		
swapSystemFileSystemMaxSize		
swapSystemFileSystemReservedSize		
swapSystemFileSystemMountPoint		

#### ( 11 )cpuUtil グループ

SNMP エージェントは HP-UX (IPF) の cpuUtil グループをサポートしていません。

cpuUtil グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-67 日立企業固有 MIB オブジェクトの実装状況 ( cpuUtil グループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.cpuUtil )  
 ( 1.116.5.1.2.1.10 )

オブジェクト名	MIB オペレーション						
	Solaris の場合		AIX の場合		Linux , Linux (IPF) の場合		
	get	set	get	set	get	set	
cpuUtilTable. cpuUtilEntry	cpuUtilNum		-		-		-
	cpuUtilUser		-		-		-
	cpuUtilSystem		-		-		-
	cpuUtilWio		-		-	-	-
	cpuUtilIdle		-		-		-
	cpuUtilTime		-		-		-
cpuUtilInterval		-		-		-	
cpuUtilTotalUser		-		-		-	
cpuUtilTotalSystem		-		-		-	

オブジェクト名	MIB オペレーション					
	Solaris の場合		AIX の場合		Linux , Linux (IPF) の場合	
	get	set	get	set	get	set
cpuUtilTotalWio		-		-	-	-
cpuUtilTotalIdle		-		-		-

## ( 12 ) virtualMemory64 グループ

SNMP エージェントは、AIX の virtualMemory64 グループをサポートしています。

virtualMemory64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-68 日立企業固有 MIB オブジェクトの実装状況 ( virtualMemory64 グループ )  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory64 )  
( 1.116.5.1.2.1.13 )

オブジェクト名	MIB オペレーション	
	AIX の場合	
	get	set
vm64PageSize		-
vm64DaemonfreePages		-
vm64Interruptions		-
vm64PageInPages		-
vm64PageOutPages		-
vm64PageReclaims		-
vm64TLBFlashes	-	-
vm64DaemonScanPages	-	-
vm64ContextSwitches		-
vm64SystemCalls		-
vm64XFileSystemFreelistPages	-	-
vm64XSwapDeviceFreeListPages	-	-
vm64FreeMemoryPages	-	-
vm64TotalSwapIns	-	-
vm64TotalSwapOuts	-	-
vm64TotalDaemonFreePages	-	-
vm64TotalDemandLoadPages	-	-

#### 4. MIB オブジェクト一覧

オブジェクト名	MIB オペレーション	
	AIX の場合	
	get	set
vm64TotalPageFaults	-	-
vm64TotalInterruptions		-
vm64TotalIntransitPageFaults	-	-
vm64TotalDemandLoadCreatePages	-	-
vm64TotalZeroFillCreatePages	-	-
vm64TotalFreeListReclaimedPages	-	-
vm64TotalPageIns		-
vm64TotalPageOuts		-
vm64TotalPageInPages	-	-
vm64TotalPageOutPages	-	-
vm64TotalSwapInPages	-	-
vm64TotalSwapOutPages	-	-
vm64TotalDaemonTicksNum	-	-
vm64TotalContextSwitches		-
vm64TotalSystemCalls		-
vm64TotalTraps		-
vm64TotalXFileSystemFreeListPages	-	-
vm64TotalXSwapDeviceFreeListPages	-	-
vm64TotalDemandZeroFillPages	-	-
vm64TotalDaemonScanPages	-	-
vm64TotalReclaimedPages	-	-
vm64TotalDeficitPages	-	-
vm64TotalReadChars	-	-
vm64TotalWriteChars	-	-
vm64TotalForks		-
vm64TotalForkPages	-	-
vm64TotalDiskBlockReads	-	-
vm64TotalDiskBlockWrites	-	-
vm64TotalProcessSwapOuts	-	-
vm64TotalSwapOutProcesses	-	-

## ( 13 ) systemInfo64 グループ

SNMP エージェントは HP-UX (IPF) の systemInfo64 グループをサポートしています。

systemInfo64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-69 日立企業固有 MIB オブジェクトの実装状況 ( systemInfo64 グループ )  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.systemInfo64 )  
( 1.116.5.1.2.1.14 )

オブジェクト名	MIB オペレーション	
	HP-UX (IPF) の場合	
	get	set
systemInfo64TTYMajor		-
systemInfo64TTYMinor		-
systemInfo64BootTime		-
systemInfo64ActiveProcessors		-
systemInfo64MaxProcessors		-
systemInfo64MaxProcesses		-
systemInfo64RunQueProcesses		-
systemInfo64XferWaitProcesses		-
systemInfo64PageInWaitProcesses		-
systemInfo64SleepProcesses		-
systemInfo64SwapOutProcesses		-
systemInfo64PhysicalMemorySize		-
systemInfo64PhysicalMemoryFreeSize		-
systemInfo64VirtualMemoryProcessSize		-
systemInfo64VirtualMemoryWaitProcessSize		-
systemInfo64PhysicalMemoryProcessSize		-
systemInfo64PhysicalMemoryWaitProcessSize		-
systemInfo64CPUStates		-
systemInfo64OpenLogicalVolumes		-
systemInfo64OpenLogicalVolumeGrps		-
systemInfo64AllocPBUFs		-
systemInfo64UsedPBUFs		-
systemInfo64MaxPBUFs		-
systemInfo64ActiveProcessEntries		-

#### 4. MIB オブジェクト一覧

オブジェクト名	MIB オペレーション	
	HP-UX (IPF) の場合	
	get	set
systemInfo64ActiveInodeEntries		-
systemInfo64ActiveFileEntries		-

#### ( 14 ) virtualMemory64Ex グループ

SNMP エージェントは HP-UX (IPF) の virtualMemory64Ex グループをサポートしています。

virtualMemory64Ex グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-70 日立企業固有 MIB オブジェクトの実装状況 ( virtualMemory64Ex グループ )  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory64Ex )  
( 1.116.5.1.2.1.15 )

オブジェクト名	MIB オペレーション	
	HP-UX (IPF) の場合	
	get	set
vm64ExPageSize		-
vm64ExDaemonfreePages		-
vm64ExInterruptions		-
vm64ExPageInPages		-
vm64ExPageOutPages		-
vm64ExPageReclaims		-
vm64ExTLBFlashes		-
vm64ExDaemonScanPages		-
vm64ExContextSwitches		-
vm64ExSystemCalls		-
vm64ExXFileSystemFreelistPages		-
vm64ExXSwapDeviceFreeLlistPages		-
vm64ExFreeMemoryPages		-
vm64ExTotalSwapIns		-
vm64ExTotalSwapOuts		-
vm64ExTotalDaemonFreePages		-
vm64ExTotalDemandLoadPages		-

オブジェクト名	MIB オペレーション	
	HP-UX (IPF) の場合	
	get	set
vm64ExTotalPageFaults		-
vm64ExTotalInterruptions		-
vm64ExTotalIntransitPageFaults		-
vm64ExTotalDemandLoadCreatePages		-
vm64ExTotalZeroFillCreatePages		-
vm64ExTotalFreeListReclaimedPages		-
vm64ExTotalPageIns		-
vm64ExTotalPageOuts		-
vm64ExTotalPageInPages		-
vm64ExTotalPageOutPages		-
vm64ExTotalSwapInPages		-
vm64ExTotalSwapOutPages		-
vm64ExTotalDaemonTicksNum		-
vm64ExTotalContextSwitches		-
vm64ExTotalSystemCalls		-
vm64ExTotalTraps		-
vm64ExTotalXFileSystemFreeListPages		-
vm64ExTotalXSwapDeviceFreeListPages		-
vm64ExTotalDemandZeroFillPages		-
vm64ExTotalDaemonScanPages		-
vm64ExTotalReclaimedPages		-
vm64ExTotalDeficitPages		-
vm64ExTotalReadChars		-
vm64ExTotalWriteChars		-
vm64ExTotalForks		-
vm64ExTotalForkPages		-
vm64ExTotalDiskBlockReads		-
vm64ExTotalDiskBlockWrites		-
vm64ExTotalProcessSwapOuts		-
vm64ExTotalSwapOutProcesses		-

4. MIB オブジェクト一覧

( 15 ) process64 グループ

SNMP エージェントは HP-UX (IPF) の process64 グループをサポートしています。

process64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-71 日立企業固有 MIB オブジェクトの実装状況 ( process64 グループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.process64 )  
 ( 1.116.5.1.2.1.16 )

オブジェクト名		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
process64Num			-
process64Table.process64Entry. ~	process64ID		-
	process64Index		-
	process64Status		-
	process64StatusFlags		-
	process64UserID		-
	process64SavedUserID		-
	process64ParentID		-
	process64GroupID		-
	process64CPUUtilization		-
	process64Priority		-
	process64CPUNice		-
	process64Processor		-
	process64StartTime		-
	process64PhysicalMemoryTextSize		-
	process64PhysicalMemoryDataSize		-
	process64PhysicalMemoryStackSize		-
	process64PhysicalMemorySharedMemorySize		-
	process64PhysicalMemoryMemoryMappedSize		-
process64PhysicalMemoryUserSize		-	
process64PhysicalMemoryIOSize		-	
process64VirtualMemoryTextSize		-	
process64VirtualMemoryDataSize		-	

オブジェクト名	MIB オペレーション	
	HP-UX (IPF) の場合	
	get	set
process64VirtualMemoryStackSize		-
process64VirtualMemorySharedMemorySize		-
process64VirtualMemoryMemoryMappedSize		-
process64VirtualMemoryUserSize		-
process64VirtualMemoryIOSize		-
process64ResidentSize		-
process64Address		-
process64SleepAddress		-
process64UserTime		-
process64SystemTime		-
process64TTYMajor		-
process64TTYMinor		-
process64Command		-
process64Executable		-
process64ResidentTime		-
process64CPUTimeTicks		-
process64TotalCPUTimeTicks		-
process64FssID		-
process64ResidentTimeCPU		-
process64MinorFaults		-
process64MajorFaults		-
process64SwapOuts		-
process64Signals		-
process64ReceivedMessages		-
process64SentMessages		-
process64MaxResidentSize		-
process64User		-

#### ( 16 ) processor64 グループ

SNMP エージェントは HP-UX (IPF) の processor64 グループをサポートしています。

#### 4. MIB オブジェクト一覧

processor64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-72 日立企業固有 MIB オブジェクトの実装状況 (processor64 グループ)  
 (enterprises.hitachi.systemExMib.CometMibs.systems.hiux.processor64)  
 (1.116.5.1.2.1.17)

オブジェクト名		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
processor64Num			-
processor64Table.processor64Entry. ~	processor64Index		-
	processor64FileSystemReadBytes		-
	processor64FileSystemWriteBytes		-
	processor64DiskBlockReadRequests		-
	processor64DiskBlockWriteRequests		-
	processor64NFSReadBytes		-
	processor64NFSWriteBytes		-
	processor64PhysicalReads		-
	processor64PhysicalWrites		-
	processor64RunQueues		-
	processor64RunQueueProcesses		-
	processor64SysExecs		-
	processor64SysReads		-
	processor64SysWrites		-
	processor64SysNamis		-
	processor64SysIgets		-
	processor64DirFileSystemReadBytes		-
	processor64SemaphoreOperations		-
	processor64MessageOperations		-
	processor64InMUXInterruptions		-
processor64OutMUXInterruptions		-	
processor64TTYRawChars		-	

オブジェクト名		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
	processor64TTYCanonChars		-
	processor64TTYOutChars		-
	processor64CPULoadAvg1		-
	processor64CPULoadAvg5		-
	processor64CPULoadAvg15		-
	processor64UserCPUTime		-
	processor64NiceCPUTime		-
	processor64SysCPUTime		-
	processor64IdleCPUTime		-
	processor64WaitCPUTime		-
	processor64BlockCPUTime		-
	processor64SwaitCPUTime		-
	processor64IntrCPUTime		-
	processor64SsysCPUTime		-

## ( 17 ) diskInfo64 グループ

SNMP エージェントは HP-UX (IPF) の diskInfo64 グループをサポートしています。

diskInfo64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-73 日立企業固有 MIB オブジェクトの実装状況 ( diskInfo64 グループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskInfo64 )  
 ( 1.116.5.1.2.1.18 )

オブジェクト ID		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
	disk64Num		-
disk64Table.disk64Entry. ~	disk64Index		-
	disk64TTYMajor		-
	disk64TTYMinor		-
	disk64BusyTimeTicks		-
	disk64Seeks		-

#### 4. MIB オブジェクト一覧

オブジェクト ID		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
	disk64Xfers		-
	disk64WordsXfers		-
	disk64WordsWriteTime		-

#### ( 18 ) swapSystem64 グループ

SNMP エージェントは HP-UX (IPF) の swapSystem64 グループをサポートしています。  
 swapSystem64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-74 日立企業固有 MIB オブジェクトの実装状況 ( swapSystem64 グループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSystem64 )  
 ( 1.116.5.1.2.1.19 )

オブジェクト ID		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
swapSystem64TotalSize			-
swapSystem64TotalEnabledSize			-
swapSystem64TotalFreeSize			-
swapSystem64TotalBlockDeviceSize			-
swapSystem64TotalBlockDeviceEnabledSize			-
swapSystem64TotalBlockDeviceFreeSize			-
swapSystem64TotalFileSystemSize			-
swapSystem64TotalFileSystemEnabledSize			-
swapSystem64TotalFileSystemFreeSize			-
swapSystem64Num			-
swapSystem64Table.swapSystem64Entry. ~	swapSystem64Index		-
	swapSystem64Place		-
	swapSystem64Flags		-
	swapSystem64Priority		-
	swapSystem64FreeSize		-
	swapSystem64BlockDeviceMajor		-

オブジェクト ID		MIB オペレーション	
		HP-UX (IPF) の場合	
		get	set
	swapSystem64BlockDeviceMinor		-
	swapSystem64BlockDeviceStartNum		-
	swapSystem64BlockDeviceSize		-
	swapSystem64FileSystemSize		-
	swapSystem64FileSystemMinSize		-
	swapSystem64FileSystemMaxSize		-
	swapSystem64FileSystemReservedSize		-
	swapSystem64FileSystemMountPoint		-

#### ( 19 ) diskBusyInfo グループ

SNMP エージェントは HP-UX (IPF) , Linux および Linux (IPF) の diskBusyInfo グループをサポートしていません。

diskBusyInfo グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-75 日立企業固有 MIB オブジェクト実装状況 ( diskBusyInfo グループ )  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusyInfo )  
( 1.116.5.1.2.1.20 )

オブジェクト名		MIB オペレーション			
		Solaris の場合		AIX の場合	
		get	set	get	set
diskBusyInfoTable.diskBusyInfoEntry ~	diskBusyInfoDeviceName		-		-
	diskBusyInfoDeviceIndex		-		-
	diskBusyInfoTime		-		-
diskBusyInfoNum			-		-

#### 注

上記の表のすべての MIB 値は、Solaris , AIX で SNMP エージェントに設定されているディスクビジー時間取得のインターバル時間 ( デフォルト 5 分 ) ごとに更新し

#### 4. MIB オブジェクト一覧

ます。そのため、Solaris、AIX のディスクビジー時間を収集する場合はインターバル時間より大きい収集間隔を設定してください。また、Solaris、AIX の場合、SNMP エージェントが起動してから最初のディスクビジー時間情報を取得するまでのインターバル時間内では、ディスクビジー時間すべての MIB 値は、noSuchName で応答します。なお、インターバル時間（分）は htc\_monagt1 のオプションで変更できます。

#### (20)fileSystem64 グループ

fileSystem64 グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-76 日立企業固有 MIB オブジェクトの実装状況 (fileSystem64 グループ)  
(enterprises.hitachi.systemExMib.CometMibs.systems.hiux.fileSystem64)  
(1.116.5.1.2.1.21)

オブジェクト名	MIB オペレーション							
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux , Linux (IPF) の場合	
	get	set	get	set	get	set	get	set
fileSystem64Mounted		-		-		-		-
fileSystem64Table	-	-	-	-	-	-	-	-
fileSystem64Entry		-		-		-		-
fileSystem64HighID1		-		-		-		-
fileSystem64LowID1		-		-		-		-
fileSystem64ID2		-		-		-		-
fileSystem64Name		-		-		-		-
fileSystem64Block		-		-		-		-
fileSystem64Bfree		-		-		-		-
fileSystem64Bavail		-		-		-		-
fileSystem64Bsize		-		-		-		-
fileSystem64Files		-		-		-		-
fileSystem64Ffree		-		-		-		-
fileSystem64Dir		-		-		-		-

#### (21)diskBusyAvail グループ

diskBusyAvail グループの日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-77 日立企業固有 MIB オブジェクトの実装状況 ( diskBusyAvail グループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusyAvail )  
 ( 1.116.5.1.2.1.22 )

オブジェクト名	MIB オペレーション							
	HP-UX (IPF) の場合		Solaris の場合		AIX の場合		Linux , Linux (IPF) の場合	
	get	set	get	set	get	set	get	set
diskBusyAvailTable	-	-		-		-	-	-
diskBusyAvailEntry	-	-	-	-	-	-	-	-
diskBusyAvailDeviceName	-	-		-		-	-	-
diskBusyAvailDeviceIndex	-	-		-		-	-	-
diskBusyAvailDiskBusy	-	-		-		-	-	-
diskBusyAvailTime	-	-		-		-	-	-
diskBusyAvailNum	-	-		-		-	-	-
diskBusyAvailInterval	-	-		-		-	-	-

## ( 22 ) AIX 固有のグループ

AIX 固有の日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-78 日立企業固有 MIB オブジェクトの実装状況 ( AIX 固有のグループ )  
 ( enterprises.hitachi.systemExMib.CometMibs.systems.aix ) ( 1.116.5.1.2.3 )

オブジェクト名		MIB オペレーション	
		get	set
system グループ (1)	systemCPUutiUser		-
	systemCPUutilSystem		-
	systemCPUutilIdle		-
	systemCPUutilWait		-
	systemActVirtualPage		-
disk グループ (2)	diskNum		-
	diskTable. diskTableEntry. ~	diskIndex	-
		diskName	-
		diskRead	-
		diskWrite	-
page グループ	pageNum		-

#### 4. MIB オブジェクト一覧

オブジェクト名			MIB オペレーション	
			get	set
(3)	pageTable. pageTableEntry. ~	pageIndex		-
		pageSpaceName		-
		pagePhysicalVol		-
		pageSize		-
		pageUsed		-
		pageActive		-
		pageAuto		-
		pageType		-

#### (23) Solaris 固有のグループ

Solaris 固有の日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-79 日立企業固有 MIB オブジェクトの実装状況 (Solaris 固有のグループ)  
(enterprises.hitachi.systemExMib.CometMibs.systems.solaris) (1.116.5.1.2.4)

オブジェクト名			MIB オペレーション	
			get	set
system グループ (1)	systemCPUUtilUser	systemCPUUtilUser		-
		systemCPUUtilSystem		-
		systemCPUUtilIdle		-
		systemCPUUtilWait		-

#### (24) Linux , Linux (IPF) 固有のグループ

Linux , Linux (IPF) 固有の日立企業固有 MIB オブジェクトの実装状況を次の表に示します。

表 4-80 日立企業固有 MIB オブジェクトの実装状況 (Linux , Linux (IPF) 固有のグループ)  
(enterprises.hitachi.systemExMib.CometMibs.systems.Linux)  
(1.116.5.1.2.5)

オブジェクト名			MIB オペレーション	
			get	set
linuxsystem グループ (1)	linuxSystemCPUUtilUser	linuxSystemCPUUtilUser		-
		linuxSystemCPUUtilSystem		-
		linuxSystemCPUUtilIdle		-

## (25) Tru64 固有のグループ

SNMP エージェントは Tru64 固有の日立企業固有 MIB オブジェクトをサポートしていません。

表 4-81 日立企業固有 MIB オブジェクトの実装状況 ( Tru64 システム固有のグループ )  
( enterprises.hitachi.systemExMib.CometMibs.systems.Tru64 )( 1.116.5.1.2.6. )

オブジェクト名		MIB オペレーション	
		get	set
Tru64system グループ (1)	tru64SystemCPUutilUser	実装されていません。	
	tru64SystemCPUutilSystem		
	tru64SystemCPUutilIdle		



# 5

## コマンドとプロセス

この章では、SNMP エージェントで使用するコマンドの機能、および文法について説明します。また、プロセスの機能、および起動オプションについて説明します。

---

コマンド

---

プロセス

---

## コマンド

---

この節では SNMP エージェントで使用するコマンドの機能、および文法について説明します。

### コマンド一覧

SNMP エージェントのコマンドを次の表に示します。

表 5-1 SNMP エージェントのコマンド

コマンド名	機能
jplesalog.sh.def	障害発生時に該当するマシンのシステム情報を採取します。
snmpcheck	マスターエージェント、およびサブエージェントの起動、停止状態を表示します。
snmpcmdchk	SNMP エージェントが MIB 値を作成するために必要とする OS コマンドのインストール状況を確認します。
snmpstart	SNMP エージェントを起動します。
snmpstop	SNMP エージェントを停止します。
snmptrap	SNMP トラップを発行します。
systemtrap	システムトラップを発行します。

注 スーパーユーザーで実行してください。

## コマンドの詳細

---

SNMP エージェントのコマンドを、次の項目から構成される共通の書式に従って説明します。ただし、すべての項目があるとは限りません。

### 形式

コマンドの形式を示します。

### 機能

コマンドの機能を説明します。

### 格納場所

コマンドの格納先ディレクトリを説明します。

### 引数

コマンドの引数を説明します。

### 実行権限

コマンドを実行するのに必要な権限を説明します。

### 戻り値

コマンドの戻り値を説明します。

### カスタマイズ

コマンド（シェルスクリプト）で、カスタマイズできる項目について説明します。

### 注意事項

コマンドについての注意事項を説明します。

SNMP エージェントの各コマンドを、アルファベット順に説明します。

## jp1esalog.sh.def

---

### 形式

jp1esalog.sh.def

### 機能

このコマンドは、障害が発生した直後に実行してください。

このコマンドは、ルートディレクトリの直下で、tar コマンドを使用して資料を採取するディレクトリまたはファイルをアーカイブしたあと、アーカイブしたデータを compress コマンドを使用して圧縮します。デフォルトでは、/tmp/jp1esa/jp1esa.log.tar.Z を作成します。ファイルがすでにあった場合、自動的に上書きします。

なお、このコマンドは、SNMP エージェントの動作中に実行できます。

### 格納場所

- AIX 以外の場合 : /opt/CM2/ESA/bin
- AIX の場合 : /usr/CM2/ESA/bin

### 引数

なし

### 実行権限

スーパーユーザーで実行してください。

### 戻り値

0 : 正常終了。

8 : 次に示すどれかの実行時エラーが発生 ( エラーメッセージが出力される )。

- スーパーユーザー権限以外で実行。
- ワーク用ディレクトリが作成不可。
- アーカイブファイルが作成不可。
- ファイル出力ディレクトリに書き込み権限なし。
- ファイル出力ディレクトリが作成不可。

### カスタマイズ

このコマンドはシェルスクリプトです。次の項目をカスタマイズできます。

デフォルトの出力ファイルを変更したい場合

次に示す行をカスタマイズします。

```
# Log output file defaultname  
OUTFILE="/tmp/jp1esa/jp1esa.log"
```

ファイル名を指定した場合、次のファイルが作成されます。  
指定したファイル名.Z

作業用ディレクトリを変更したい場合

次に示す行をカスタマイズします。

```
# Working directory
WORKDIR="/tmp/jplesa/work"
```

このコマンド実行時には、一次作業用ディレクトリが作成され、そこに一時的にファイルが作成されるため、ある程度のディスク容量が必要になります。デフォルトでは出力ファイルと同じディレクトリに作成されますが、変更したい場合は上記の行を変更してください。

ユーザー作成ファイルなど、ほかの資料を採取したい場合

次に示す行をカスタマイズします。

```
# User Additional files
ADDFILE=""
```

なお、変数 ADDFILE に、採取する資料を追加する場合は、必ずフルパスの前に「.(ピリオド)」を付けて指定してください。また、複数指定する場合は半角スペースで区切って指定してください。

(例) ADDFILE="/var/tmp/user\_log ./etc/opt/sample/conf"

### 注意事項

- このコマンドは jp1esalog.sh.def として提供します。このファイルを適当なディレクトリにコピーして使用してください。この製品はインストール時に毎回 jp1esalog.sh.def を上書きするため、jp1esalog.sh.def をカスタマイズした場合は、カスタマイズした内容が削除されます。
- 採取したデータを ftp など転送する際はバイナリーモードで転送してください。
- 資料を採取するマシンに compress コマンドがインストールされていない場合は、圧縮はしません。/tmp/jplesa/jplesa.log.tar ファイルを作成します。
- jp1esalog.sh.def コマンドで取得する資料の一覧を次の表に示します。

表 5-2 jp1esalog.sh.def コマンドで取得する資料の一覧

分類	取得資料
共通情報	/etc/hosts
	/etc/services
	シスログファイル • HP-UX (IPF) /var/adm/syslog/syslog.log /var/adm/syslog/OLDsyslo.log • Solaris /var/adm/messages • Linux , Linux (IPF) /var/log/messages

分類	取得資料
	ps -ef の結果 ( Solaris 10 の場合は ps -lf -z ゾーン名 )
	ps -e の結果 ( AIX だけ )
	netstat ( -a , -i , -rv , -an ) の結果
	uname -a の結果
	hostname の結果
	id の結果
	env の結果
	bdf -l または df -k の結果
	/etc/.hitachi/pplistd/pplistd
	<p>ネイティブエージェントの構成ファイルおよびログファイル Solaris</p> <ul style="list-style-type: none"> <li>• /etc/snmp/conf/snmpd.conf</li> </ul> <p>Solaris10 では次のファイルも取得します。</p> <ul style="list-style-type: none"> <li>• /etc/sma/snmp/snmpd.conf</li> <li>• /etc/init.d/init.sma</li> <li>• /var/log/snmpd.log</li> </ul> <p>SMF 機能に対応した Solaris10 では次のファイルも取得します。</p> <ul style="list-style-type: none"> <li>• /etc/release</li> <li>• /lib/svc/method/svc-sma</li> <li>• /lib/svc/method/svc-snmpdx</li> <li>• /var/svc/log/application-management-snmpdx:default.log</li> <li>• /var/svc/log/application-management-sma:default.log</li> <li>• /var/svc/log/milestone-multi-user:default.log</li> <li>• /var/opt/CM2/ESA/log/initdesa.log.err</li> <li>• /var/opt/CM2/ESA/log/snmpstart.log.err</li> </ul> <p>AIX</p> <ul style="list-style-type: none"> <li>• /etc/snmpd.conf</li> <li>• /usr/tmp/snmpd.log</li> <li>• /etc/snmpdv3.conf</li> <li>• /usr/tmp/snmpdv3.log</li> </ul> <p>Linux , Linux (IPF)</p> <ul style="list-style-type: none"> <li>• /etc/snmp/snmpd.conf</li> <li>• /var/log/snmpd.log</li> </ul>
SNMP エージェントの情報	<p>/var/adm/ 配下にある次に示すファイル</p> <ul style="list-style-type: none"> <li>• snmpd.log<sub>n</sub> <sup>1, 2</sup> n : ログファイルの面数</li> </ul> <hr/> <ul style="list-style-type: none"> <li>• /tmp/jp1esa/work/jp1esalog_err.log</li> </ul> <hr/> <ul style="list-style-type: none"> <li>• /var/opt/CM2/ESA/log/htc_monagt1.log ( Solaris , Linux )</li> <li>• /usr/CM2/ESA/log/htc_monagt1.log ( AIX )</li> </ul> <hr/> <p>/opt/CM2/ESA/bin/snmpcheck の結果</p>

分類	取得資料
	/etc/SnmpAgent.d の配下
	/etc/srconf/agt の配下
	<ul style="list-style-type: none"> <li>• /opt/CM2/ESA/opt の配下 ( HP-UX (IPF) , Linux , Linux (IPF) )</li> <li>• /etc/rc.config.d/Snmp で始まるファイル ( Solaris )</li> <li>• /usr/CM2/ESA/opt の配下 ( AIX だけ )</li> </ul>
	<ul style="list-style-type: none"> <li>• /opt/CM2/ESA/ext の配下 ( AIX 以外 )</li> <li>• /usr/CM2/ESA/ext の配下 ( AIX だけ )</li> </ul>
	/core ( 存在する場合 )
	/root/core ( Linux , Linux (IPF) の場合で存在する場合 )

注 1 snmpd.logn ファイルにはログ・16 進数ダンプ・VarBind リストが出力されま  
す。snmpd.logn ファイルのサイズと面数は、SnmpMaster ファイルの次の環境変数で指  
定します。

- ファイルサイズ：SNMP\_HTC\_SNMPD\_LOG\_SIZE

SNMP\_HTC\_SNMPD\_LOG\_SIZE 環境変数の指定例を次に示します。単位はメガ  
バイトです。例では 10 メガバイトを指定しています。

(例)

```
SNMP_HTC_SNMPD_LOG_SIZE=10
export SNMP_HTC_SNMPD_LOG_SIZE
```

- ファイルの面数：SNMP\_HTC\_SNMPD\_LOG\_CNT

SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数の指定例を次に示します。単位は面数で  
す。例では 10 面を指定しています。

(例)

```
SNMP_HTC_SNMPD_LOG_CNT=10
export SNMP_HTC_SNMPD_LOG_CNT
```

環境変数の詳細については、「snmpdm」を参照してください。

また、デフォルトではファイル 1 面当たり 10 メガバイトで、10 面作成するため  
snmpd.logn の格納先には 100 メガバイトを格納できる領域が必要です。

注 2 snmpd.logn ファイルに出力されるプロセスは次のとおりです。

```
snmpdm , naaagt , hp_unixagt , extsubagt , trapdestagt , htc_unixagt1 ,
htc_unixagt2 , htc_unixagt3
```

## snmpcheck

---

### 形式

snmpcheck

### 機能

マスターエージェント、サブエージェントおよびネイティブエージェントの起動、停止状態を表示します。

### 表示例

Solaris で snmpcheck コマンドを実行した場合の例を次に示します。

```
#/opt/CM2/ESA/bin/snmpcheck
snmpdm  running pid=15128
hp_unixagt      running pid=15170
trapdestagt    running pid=15189
extsubagt      not running
htc_unixagt1   running pid=15209
htc_unixagt3   running pid=15229
htc_monagt1    running pid=15248
naaagt  running pid=15151
snmpdx  running pid=15132
```

「running」はプロセスが起動していることを表します。

「not running」はプロセスが未起動であることを表します。

「pid」はプロセス ID を表します。

### 格納場所

- AIX 以外の場合 : /opt/CM2/ESA/bin
- AIX の場合 : /usr/CM2/ESA/bin

### 引数

なし。

# snmpcmdchk

---

## 形式

snmpcmdchk

## 機能

このコマンドは、SNMP エージェントが MIB 値を作成するために必要とする OS のコマンドがインストールされているかどうかを確認するコマンドです。SNMP エージェントをインストール後、このコマンドを実行し、インストールされていないコマンドがある場合は、インストールする必要があります。

## 指定例

Solaris で snmpcmdchk コマンドを実行した場合の例を次に示します。

```
#/opt/CM2/ESA/bin/snmpcmdchk
/etc/prtconf          installed.
/usr/bin/sar          installed.
/etc/swap             installed.
/usr/bin/pagesize     installed.
/usr/bin/mpstat       Not installed.
```

「installed」は、コマンドがインストールされていることを示します。

「Not installed」は、コマンドがインストールされていないことを示します。

## 格納場所

- AIX 以外の場合：/opt/CM2/ESA/bin
- AIX の場合：/usr/CM2/ESA/bin

## 引数

なし。

# snmpstart

---

## 形式

snmpstart [-e]

## 機能

SNMP エージェントを起動します。snmpstart コマンド実行時、SNMP エージェントが起動されている場合、snmpstart コマンドは SNMP エージェントを停止します。そのあと、SNMP エージェントを再起動します。

## 格納場所

- AIX 以外の場合 : /opt/CM2/ESA/bin
- AIX の場合 : /usr/CM2/ESA/bin

## 引数

-e

/opt/CM2/ESA/ext ディレクトリの下にあり、かつ読み込まれていない拡張 MIB 定義ファイルを読み込みます。

## 実行権限

スーパーユーザーで実行してください。

# snmpstop

---

## 形式

snmpstop

## 機能

SNMP エージェントを停止します。

## 格納場所

- AIX 以外の場合 : opt/CM2/ESA/bin
- AIX の場合 : /usr/CM2/ESA/bin

## 引数

なし。

## 実行権限

スーパーユーザーで実行してください。

## snmptrap

---

### 形式

```
snmptrap [-d] [-p ポート番号] [-c コミュニティ名]  
          ノード名 企業ID agentアドレス 標準トラップ番号  
          拡張トラップ番号 タイムスタンプ  
          [オブジェクト識別子 値のシンタクス 値...]
```

### 機能

指定ノードに対して SNMP トラップを発行します。

### 格納場所

- AIX 以外の場合 : /opt/OV/bin
- AIX の場合 : /usr/OV/bin

### 引数

-d

SNMP パケットを 16 進数表示, およびデコーディングされた ASN.1 形式で標準出力に出力します。

-p ポート番号

送信するマネージャーのポート番号を指定します。省略時すると 162 が仮定されます。

-c コミュニティ名

コミュニティ名を指定します。省略すると public が仮定されます。

ノード名

IP アドレス, またはホスト名を指定します。

企業 ID

sysObjectID を指定します。形式は A.B.C.D... です。ただし A, B, C, D は 10 進数表示のサブ識別子です。形式 A.B.C.D... ではオブジェクト識別子の先頭が '1.3.6.1.2.1' であれば, '1.3.6.1.2.1' は省略できます。null 文字列 (") は SNMP エージェントの sysObjectID になります。SNMP エージェント機能の sysObjectID については, 「4.3.1 日立企業固有 MIB オブジェクトの構成」を参照してください。

agent アドレス

IP アドレス, またはホスト名を指定します。NULL 文字列 (") を指定した場合, 該当するシステムの OS の関数を使用して取得したホスト名を, OS の関数を使用して IP アドレスに変換した値になります。

なお、snmptrap コマンドで発行するトラップメッセージ中のエージェントアドレスに、特定の IP アドレスを使用したい場合は、「agent アドレス」にその IP アドレスを指定してください。

#### 標準トラップ番号

標準トラップ番号を 0 ~ 6 の整数で指定します。拡張トラップを発行する場合は 6 を指定します。

#### 拡張トラップ番号

拡張トラップ番号を 32 ビットの整数で指定します。ただし、標準トラップ番号が 6 でない場合無視され 0 が埋められます。有効値は正数、負数、16 進数 (0x ~)、および 8 進数 (0 ~) です。

#### タイムスタンプ

時刻を 0 以上の整数で指定します。NULL 文字列 (" ) を指定した場合にはシステム開始からの timeticks が仮定されます。

#### オブジェクト識別子

snmptrap コマンドは、「オブジェクト識別子、値のシンタクス、値」を 1 組とした引数を複数指定できます。一度に指定できる引数の数は一つの値の長さを 256 バイトとした場合、最大 20 個を目安としてください。オブジェクト識別子の形式は A.B.C.D... です。A, B, C, D は 10 進数表示のサブ識別子です。形式 A.B.C.D... ではオブジェクト識別子の先頭が '1.3.6.1.2.1' であれば、'1.3.6.1.2.1' は省略できます。

#### 値のシンタクス

値のシンタクスは次の中から選択します。

integer (  $-2^{31} \sim 2^{31}-1$  ), octetstring, objectidentifier (形式が A.B.C.D... で、オブジェクト識別子の先頭が '1.3.6.1.2.1' であれば、'1.3.6.1.2.1' は省略できます), null (このあとの '値' に何を設定しても無視されます), ipaddress, counter (0 ~ 4294967295), gauge, timeticks, opaque。それぞれの詳細については RFC1155 を参照してください。

また、octetstring の特別なケースとして、octetstringhex (00 ~ FF の 16 進数列指定、例: 01FF), octetstringoctal (000 ~ 377 の 8 進数列指定、例: 001377), octetstringascii (ASCII コード文字列指定) が指定できます。opaque の特別なケースとして、opaquehex (00 ~ FF の 16 進数列指定、例: 01FF), opaqueoctal (000 ~ 377 の 8 進数列指定、例: 001377), opaqueascii (ASCII コード文字列指定) が指定できます。

#### 値

値のシンタクスに対して正しい値を指定します。

## 戻り値

### 0: 正常終了

SNMPトラップはUDPで送信するので、相手ノードとの送達確認はしません。そのため、正常終了した場合でも相手ノードに通知されない場合があります。

### 1: 実行時エラー

エラーメッセージを出力します。

## 注意事項

AIXでcronや/etc/inittabから起動されたシェルスクリプトやプログラムの延長でsnmptrapコマンドを実行した場合、次のようなエラーが出力されコマンドが失敗することがあります。

```
snmptrap:cannot set locale($LANG="Ja_JP")
```

この場合、環境変数LC\_ALLに使用する言語を設定してください。

使用する言語に'C'を設定する場合のBシェルの例を次に示します。

```
LC_ALL=C  
export LC_ALL  
snmptrap flcmdmak .1.3.6.1.4.1.4242 15.6.71.223 6 2 0
```

# systemtrap

---

## 形式

systemtrap

## 機能

指定されたノードに対し、システムトラップを発行します。

このコマンドの詳細については、マニュアル「JP1/Cm2/SNMP System Observer」を参照してください。

systemtrap コマンドは、監視プログラムとして JP1 Version 8 以前の JP1/Cm2/SNMP System Observer を使用する場合の旧バージョン互換のために提供しています。

systemtrap コマンドの詳細については、JP1 Version 8 以前の JP1/Cm2/SNMP System Observer のマニュアルを参照して下さい。

## プロセス

この節では SNMP エージェントのプロセスの機能および起動オプションについて説明します。また、プロセスで使用する環境変数についても説明します。

### プロセス一覧

SNMP エージェントはマスターエージェントとサブエージェントで構成されます。また、OS から情報を収集するための情報収集デーモンも動作します。

SNMP エージェントで動作するプロセスを次の表に示します。

#### マスターエージェントで動作するプロセス

表 5-3 SNMP エージェントのプロセス (マスターエージェント)

コマンド名	機能
snmpd	マスターエージェントへの各種指示をします。

#### サブエージェントで動作するプロセスおよび情報収集デーモン

表 5-4 SNMP エージェントのプロセス (サブエージェント・情報収集デーモン)

コマンド名	機能
extsubagt	拡張 MIB オブジェクトを提供するサブエージェントです。
hp_unixagt	hp-unix MIB を提供するサブエージェントです。
htc_monagt1	CPU 利用率を定期収集するデーモンプロセスです。
htc_unixagt1	日立企業固有 MIB (SystemInfo グループ, VirtualMemory グループ, Process グループ) を提供するサブエージェントです。
htc_unixagt2	日立企業固有 MIB (Processor グループ, DiskInfo グループ, SwapInfo グループ) を提供するサブエージェントです。
htc_unixagt3	日立企業固有 MIB (cpuUtil グループ, VirtualMemory64 グループ) を提供するサブエージェントです。
naaagt	ネイティブエージェントアダプター機能を提供するサブエージェントです。
trapdstagt	HP 企業固有 MIB (Trap グループ) を提供するサブエージェントです。

## プロセスの詳細

---

SNMP エージェントのプロセスを、次の項目から構成される共通の書式に従って説明します。ただし、すべての項目があるとは限りません。

### 形式

プロセス起動の形式を示します。

### 機能

プロセスの機能を説明します。

### 格納場所

プロセスの格納先ディレクトリを説明します。

### 引数

プロセスの起動オプションを説明します。

### 外部の影響

環境変数を説明します。

### 注意事項

プロセスについての注意事項を説明します。

## snmpdm

---

### 形式

```
snmpdm [ -authfail ] [ -Contact 連絡先 ] [ -help ] [ -Location 設置場所 ]  
        [ -mask ログマスク値 ] [ -n ] [ -tcplocal ]  
        [ -sysDescr 説明 ] [ -hexdump ] [ -vbdump ]
```

### 機能

マスターエージェントへの各種指示をします。マスターエージェントは、SNMP マネージャーからの SNMP 要求を受信します。その受信したメッセージをサブエージェントに通知します。サブエージェントは、応答をマスターエージェントに返信します。マスターエージェントは、その応答を SNMP マネージャーに返信します。

マスターエージェントは次に示す MIB を提供します。

- System グループ ( internet.mgmt.mib-2.system )
- SNMP グループ ( internet.mgmt.mib-2.snmp )
- SnmpdConf グループ ( enterprises.hp.nm.snmp.snmpdConf )

### 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

### 引数

-authfail

-authfail の省略形 : -a

マスターエージェントの認証失敗トラップの送信を抑制します。

この引数は、前バージョンの SNMP エージェントとの互換性のために用意されているだけなので、通常は使用しないでください。

認証失敗トラップを抑制する場合は、`/etc/srconf/agt/snmpd.cnf` 内の `snmpEnableAuthenTraps` に 2 を設定し、マスターエージェントを再起動してください。

-a を指定してマスターエージェントを起動すると、マスターエージェントは `/etc/srconf/agt/snmpd.cnf` 内の `snmpEnableAuthenTraps` の設定を 2 に書き換えます。そのあと、認証失敗トラップを抑制したくない場合は、`/etc/srconf/agt/snmpd.cnf` 内の `snmpEnableAuthenTraps` に 1 を設定し、マスターエージェントを再起動してください。

-Contact 連絡先

-Contact の省略形 : -C

マスターエージェントの連絡先を変更します。

-help

-help の省略形：-h

マスターエージェントのコマンドの引数を表示します。

-Location 設置場所

-Location の省略形：-L

マスターエージェントの設置場所を変更します。

-mask ログマスク値

-mask の省略形：-m

マスターエージェントのログマスク値を、指定されたログマスク値に変更します。

ログマスク値は、文字列、10 進数、および 16 進数で指定できます。ログマスク値の指定と指定例を次に示します。

値の種別	ログの抑止	トレースログ出力	警告ログ出力	エラーログ出力
文字列	-	FACTORY_TRACE	FACTORY_WARN	FACTORY_ERROR
10 進数	0	8388608	268435456	536870912
16 進数	0x0	0x00800000	0x10000000	0x20000000

( 凡例 )

- : 該当しない。

指定例

```
snmpdm -m FACTORY_TRACE
snmpdm -m 8388608
snmpdm -m 0x00800000
snmpdm -m FACTORY_TRACE FACTORY_WARN FACTORY_ERROR
```

-m を文字列で指定する場合で、かつ、ほかの引数と組み合わせて指定する場合は、-m は最後に指定してください。

-n

マスターエージェントは、デーモン化しません。

-tcplocal

サブエージェントからの TCP 接続を受け付けます。

-sysDescr 説明

-sysDescr の省略形：-sys

マスターエージェントの sysDescr を変更します。

-hexdump

SNMP パケットの内容を 16 進数ダンプで表示します（使用方法については「6.4.1 マスターエージェントの送受信パケットダンプの取得」を参照してください）。

-vbdump

SNMP パケット中の VarBind リストの内容を表示します（使用方法については「6.4.1 マスターエージェントの送受信パケットダンプの取得」を参照してください）。

-apverbose

冗長ログを出力します。

## 外部の影響

### 環境変数

次の環境変数を SnmpMaster ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。マスターエージェントは、この環境変数が指定されていない場合、SNMP 受信ポートに /etc/services ファイルの snmp 行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合は、指定する必要があります。

#### SNMP\_HTC\_AUTH\_LOG

ログの取得を指定します。

SNMP\_HTC\_AUTH\_LOG 環境変数の指定方法については、「6.4.4 コミュニティ名不正時のログ取得」を参照ください。

#### SR\_TRAP\_TEST\_PORT

マスターエージェントの SNMP トラップの通知先ポートを指定します。

#### SNMP\_HTC\_INIT\_WAIT\_TIME

snmpdm プロセスが起動されたあと coldStart トラップを送信するまでの時間を 0 ~ 300（単位は秒）で指定します。この間にきた SNMP Request は破棄されます。デフォルトは 15 秒です。

#### SNMP\_HTC\_SNMPD\_LOG\_SIZE

ログ・16 進数ダンプ・VarBind リストの出力ファイルのサイズを 0 ~ 50（単位はメガバイト）で指定します。なお、0 を指定した場合、ログ・16 進数ダンプ・VarBind リストは出力されません。デフォルトは 10 メガバイトです。

#### SNMP\_HTC\_SNMPD\_LOG\_CNT

ログ・16 進数ダンプ・VarBind リストの出力ファイルの面数を 1 ~ 10 で指定しま

す。デフォルトは 10 面です。

#### SR\_LOG\_DIR

ログ・16 進数ダンプ・VarBind リストの出力先ディレクトリを指定します。

#### 注意事項

- SNMP\_HTC\_INIT\_WAIT\_TIME 環境変数で指定する時間に 15 秒より短い値に設定する場合、マネージャーからの要求に対して noSuchName エラーが返される可能性が高くなります。

## extsubagt

---

### 形式

```
extsubagt [-e 拡張MIB定義ファイル] [-p]
          [-E プライオリティ] [-aperror] [-apwarn] [-aptrace]
          [-apconfig] [-appacket] [-aptrap] [-apaccess]
          [-apemanate] [-apverbose] [-apuser] [-apall] [-help]
          [-retry N] [-fcmdguard N] [-pipeguard N]
          [-invokeid]
```

### 機能

拡張 MIB オブジェクトを提供します。

### 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

### 引数

-e 拡張 MIB 定義ファイル

拡張 MIB 定義ファイルを指定します。この引数を省略した場合、/etc/SnmpAgent.d/snmpd.extend が使用されます。

-p

拡張 MIB 定義ファイルのシンタクスチェックをします。

-E プライオリティ

サブエージェントのプライオリティを指定します。

-aperror

エラーログを出力します。

-apwarn

エラーおよび警告ログを出力します。

-aptrace

トレースログを出力します。

-apconfig

構成ファイルに関するログを出力します。

-appacket

パケットの組み立ておよび解析ログを出力します。

-aptrap

トラップメッセージに関するログを出力します。

-apaccess

エージェント処理に関するログを出力します。

-apemanate

マスターエージェントとサブエージェントに関するログを出力します。

-apverbose

冗長ログを出力します。

-apuser

ユーザーログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-retry N

マスターエージェントの接続を、N 秒単位で実行します。

-fcmdguard N

file\_command の実行応答の監視時間を、N 秒単位 (1 N 90) で設定します。

-pipeguard N

SNMP エージェントが、pipe\_out\_name にデータを書き込んでから処理結果が書き込まれるまでの監視時間を、N 秒単位 (1 N 90) で設定します。

-invokeid

pipe\_out\_name に書き込む引数の先頭に識別番号を付けます。識別番号の形式は、xxxxxxx.yyyyyy (xxxxxxx: 通算秒,yyyyyy: マイクロ秒) です。

-fcmdguard, -pipeguard, -invokeid が、起動時または snmpstart コマンド実行時に、常に有効になるように設定できます。詳細については、「3.1.1 SNMP エージェントの起動」を参照してください。

## 外部の影響

### 環境変数

次の環境変数を SnmpExtagt ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合はマスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は `/etc/services` ファイルの `snmp` の行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

## hp\_unixagt

---

### 形式

```
hp_unixagt  [ -aperror ] [ -apwarn ] [ -aptrace ] [ -apconfig ]  
            [ -appacket ] [ -aptrap ] [ -apaccess ] [ -apemanate ]  
            [ -apverbose ] [ -apuser ] [ -apall ] [ -help ]  
            [ -retry N ]
```

### 機能

次に示す HP 企業固有 MIB を提供します。

- ComputerSystem グループ ( enterprises.hp.nm.system.general.computerSystem )
- FileSystem グループ ( enterprises.hp.nm.system.general.fileSystem )
- Processes グループ ( enterprises.hp.nm.system.general.processes )
- ICMP グループ ( enterprises.hp.nm.icmp )

### 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

### 引数

-aperror

エラーログを出力します。

-apwarn

エラーおよび警告ログを出力します。

-aptrace

トレースログを出力します。

-apconfig

構成ファイルに関するログを出力します。

-appacket

パケットの組み立ておよび解析ログを出力します。

-aptrap

トラップメッセージに関するログを出力します。

-apaccess

エージェント処理に関するログを出力します。

-apemanate

マスターエージェントとサブエージェントに関するログを出力します。

-apverbose

冗長ログを出力します。

-apuser

ユーザーログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-retry N

マスターエージェントの接続を、N 秒単位で実行します。

## 外部の影響

### 環境変数

次の環境変数を SnmpHpunix ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合はマスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は /etc/services ファイルの snmp の行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

#### SNMP\_HTC\_SOLARIS\_SWAP\_RESERVED (Solaris の場合)

デバイススワップ空間サイズに予約済みのサイズを含めるかどうかを指定します。

Y : 予約済みのサイズを含める。

Y 以外 : 予約済みのサイズを含めない。

デフォルトは Y 以外です。

#### SNMP\_HTC\_AIX\_EXCEPT\_FILECACHE (AIX の場合)

使用中の物理メモリー容量からファイルキャッシュを除くかどうかを指定します。

Y : 使用中の物理メモリー容量からファイルキャッシュを除く。

Y 以外 : 使用中の物理メモリー容量からファイルキャッシュを除かない。

デフォルトは Y 以外です。

# htc\_monagt1

## 形式

```
htc_monagt1 [-i CPU利用率取得のインターバル時間]
[-s CPU利用時間情報取得のインターバル時間 (Solaris, AIXの場合だけ有効)]
[-d ディスクビジー時間取得のインターバル時間 (Solaris, AIXの場合だけ有効)]
[-t トレースマスク値]
[-k]
[-T トレースマスク値]
```

## 機能

CPU 利用率, CPU 利用時間, およびディスクビジー時間を定期収集します。このコマンドはデーモンプロセスです。

オプションと MIB の対応を次に示します。

オプション	MIB
-i (CPU 利用率)	hitachi.systemExMib.cometMibs.system.hiux.cpuUtil グループ
-s (CPU 利用時間)	Solaris,AIX の場合 hp.nm.system.general.computerSystem computerSystemUserCPU computerSystemSysCPU computerSystemIdleCPU computerSystemNiceCPU hitachi.systemExMib.cometMibs.system.hiux.Processor グループ processorUserCPUtime processorNiceCPUtime processorSysCPUtime processorIdleCPUtime processorWaitCPUtime
-d (ディスクビジー時間)	hitachi.systemExMib.cometMibs.system.hiux.diskBusyInfo グループ hitachi.systemExMib.cometMibs.system.hiux.diskBusyAvail グループ

## 格納場所

- Solaris, HP-UX (IPF) の場合: /opt/CM2/ESA/bin
- AIX, Linux, Linux (IPF) の場合: /usr/sbin

## 引数

-i CPU 利用率取得のインターバル時間 ((0 ~ 1440))<<5>>

CPU 利用率取得のインターバル時間を分単位で指定します。なお, 0 を指定した場合は CPU 利用率情報を取得しません。

-s CPU 利用時間情報取得のインターバル時間 ((0 ~ 1440))<<5>>

CPU 利用時間情報のインターバル時間を分単位で設定します。なお, 0 を指定した場合

## 5. コマンドとプロセス htc\_monagt1

は CPU 利用時間情報を取得しません。

-d ディスクビジー時間取得のインターバル時間 ((0 ~ 1440))<<5>>

ディスクビジー時間取得のインターバル時間を分単位で設定します。なお、0 を指定した場合はディスクビジー時間情報を取得しません。

-t トレースマスク値

htc\_monagt1 のトレースマスク値を、指定されたトレースマスク値に変更します。

この引数が指定されていない場合、トレースマスク値は 0 となります。

トレースマスクの値を次に示します。

トレースマスク	取得内容
0	トレースの停止
1	トレースの開始

ログおよびトレースは、/var/opt/CM2/ESA/log/htc\_monagt1.log に取得します。このファイルは 4MB 以上になった場合、/var/opt/CM2/ESA/log/htc\_monagt1.log の内容を /var/opt/CM2/ESA/log/htc\_monagt1.log.old にコピーし、/var/opt/CM2/ESA/log/htc\_monagt1.log を上書きします。

-k

起動中の htc\_monagt1 に対して終了要求を送信します。

-T トレースマスク値

起動中の htc\_monagt1 に対してトレースマスクの変更を通知します。

### 外部の影響

#### 環境変数

次の環境変数を SnmpHtcMonagt1 ファイルで指定します。

#### SNMP\_HTCMONAGT1\_START

SNMP エージェント起動時に htc\_monagt1 を起動するかどうかを指定します。

Y：起動する。

N：起動しない。

#### SNMP\_HTC\_AIX\_CPU\_SMT (AIX の場合)

CPU の利用率の取得方法を指定します。

Y：マシン全体の CPU 利用率を取得する。

Y 以外：CPU 個々の CPU 利用率を取得する。

デフォルトは Y 以外です。

### 注意事項

- `-i`, `-s`, `-d` オプションがすべて 0 の場合, `htc_monagt1` は起動しません。
- `SNMP_HTC_AIX_CPU_SMT` 環境変数で Y を指定した場合, CPU 個々の CPU 利用率は監視できなくなります。

## htc\_unixagt1

---

### 形式

```
htc_unixagt1  [-aperror] [-apwarn] [-aptrace] [-apconfig]  
              [-appacket] [-aptrap] [-apaccess]  
              [-apemanate] [-apverbose] [-apuser] [-apall]  
              [-help] [-retry N]
```

### 機能

次に示す日立企業固有 MIB を提供します。

- SystemInfo グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.systemInfo )
- VirtualMemory グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory )
- Process グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.process )
- SwapSpace グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSpace )
- DiskBusy グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusy )
- systemInfo64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.systemInfo64 )
- virtualMemory64Ex グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory64Ex )
- process64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.process64 )
- fileSystem64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.fileSystem64 )
- system グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.aix.system )
- disk グループ ( enterprises.hitachi.systemExMib.CometMibs.systems.aix.disk )
- page グループ ( enterprises.hitachi.systemExMib.CometMibs.systems.aix.page )
- system グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.solaris.system )
- LinuxSystem グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.Linux.linuxsystem )
- Tru64system グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.Tru64.Tru64system )
- OpConf グループ  
( enterprises.hitachi.systemExMib.CometMibs.SubSystems.cometOP.OpConf )

## 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

## 引数

-aperror

エラーログを出力します。

-apwarn

エラーおよび警告ログを出力します。

-aptrace

トレースログを出力します。

-apconfig

構成ファイルに関するログを出力します。

-appacket

パケットの組み立ておよび解析ログを出力します。

-aptrap

トラップメッセージに関するログを出力します。

-apaccess

エージェント処理に関するログを出力します。

-apemanate

マスターエージェントとサブエージェントに関するログを出力します。

-apverbose

冗長ログを出力します。

-apuser

ユーザーログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-retry N

マスターエージェントの接続を、N 秒単位で実行します。

## 外部の影響

### 環境変数

次の環境変数を SnmpHtcunix1 ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合は、マスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は、`/etc/services` ファイルの `snmp` 行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

#### SNMP\_HTC\_FILE\_EXTEND

Linux, Linux (IPF) のカーネルのバージョンが 2.4.21-27.EL 以降で、ファイルシステム容量が 4 テラバイトを超える場合やファイルシステム中の `i` ノード総数が  $2^{32}-1$  個を超える場合でも、`FileSystem64Block` の MIB オブジェクトを取得するときに指定します。

## htc\_unixagt2

---

### 形式

```
htc_unixagt2  [ -aperror ] [ -apwarn ] [ -aptrace ] [ -apconfig ]  
              [ -appacket ] [ -aptrap ] [ -apaccess ]  
              [ -apemanate ] [ -apverbose ] [ -apuser ] [ -apall ]  
              [ -help ] [ -retry N ]
```

### 機能

次に示す日立企業固有 MIB を提供します。

- Processor グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.pocessor )
- DiskInfo グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskInfo )
- SwapInfo グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapInfo )
- processor64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.processor64 )
- diskInfo64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskInfo64 )
- swapSystem64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.swapSystem64 )

### 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

### 引数

-aperror

エラーログを出力します。

-apwarn

エラーおよび警告ログを出力します。

-aptrace

トレースログを出力します。

-apconfig

構成ファイルに関するログを出力します。

-appacket

パケットの組み立ておよび解析ログを出力します。

-aptrap

トラップメッセージに関するログを出力します。

-apaccess

エージェント処理に関するログを出力します。

-apemanate

マスターエージェントとサブエージェントに関するログを出力します。

-apverbose

冗長ログを出力します。

-apuser

ユーザーログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-retry N

マスターエージェントの接続を、N 秒単位で実行します。

## 外部の影響

### 環境変数

次の環境変数を SnmpHtcunix2 ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合は、マスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は、`/etc/services` ファイルの `snmp` 行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

## htc\_unixagt3

---

### 形式

```
htc_unixagt3 [-aperror] [-apwarn] [-aptrace] [-apconfig]
              [-appacket] [-aptrap] [-apaccess]
              [-apemanate] [-apverbose] [apuser] [-apall] [-help]
              [-retry N]
```

### 機能

次に示す日立企業固有 MIB を提供します。

- cpuUtil グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.cpuUtil )
- virtualMemory64 グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.virtualMemory64 )
- diskBusyInfo グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusyInfo )
- diskBusyAvail グループ  
( enterprises.hitachi.systemExMib.CometMibs.systems.hiux.diskBusyAvail )

### 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

### 引数

-aperror

エラーログを出力します。

-apwarn

エラーおよび警告ログを出力します。

-aptrace

トレースログを出力します。

-apconfig

構成ファイルに関するログを出力します。

-appacket

パケットの組み立ておよび解析ログを出力します。

-aptrap

トラップメッセージに関するログを出力します。

-apaccess

エージェント処理に関するログを出力します。

-apemanate

マスターエージェントとサブエージェントに関するログを出力します。

-apverbose

冗長ログを出力します。

-apuser

ユーザーログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-retry N

マスターエージェントの接続を、N 秒単位で実行します。

## 外部の影響

### 環境変数

次の環境変数を SnmpHtcunix3 ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合は、マスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は、`/etc/services` ファイルの `snmp` 行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

# naaagt

---

## 形式

```
naaagt [-aperror] [-apwarn] [-aptrace] [-apall] [-help]  
        [-port ポート番号] [-readcomm コミュニティ名]  
        [-timeout タイムアウト時間]  
        [-writecomm コミュニティ名]  
        [-hexdump] [-vbdump] [-n]
```

## 機能

ネイティブエージェントアダプター機能を提供します。

## 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

## 引数

-aperror

エラーログを出力します。

-apwarn

エラーおよび警告ログを出力します。

-aptrace

トレースログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-port ポート番号

ネイティブエージェントと接続するための UDP ポート番号を変更します。指定するポート番号はあて先ポート番号です。省略すると 8161 が仮定されます。

-readcomm コミュニティ名

naaagt コマンドがネイティブエージェントに対して GET リクエストを送信するときに使用するコミュニティ名を変更します。省略すると public が仮定されます。

GET リクエストで使用するコミュニティ名については「2.5.3(2) ネイティブエージェントのコミュニティ名」を参照してください。

## 5. コマンドとプロセス

naaagt

-timeout タイムアウト時間

naaagt コマンドがネイティブエージェントからの GET レスポンスを待つタイムアウト時間を変更します。タイムアウト時間は秒単位で指定します。省略すると 4 秒が仮定されます。

-writecomm コミュニティ名

naaagt コマンドがネイティブエージェントに対して SET リクエストを送信するときに使用するコミュニティ名を変更します。省略すると public が仮定されます。

)SET リクエストで使用するコミュニティ名については「2.5.3(2) ネイティブエージェントのコミュニティ名」を参照してください。

-hexdump

naaagt コマンドが送受信する SNMP パケットの内容を 16 進数ダンプで標準出力に表示します。

-vbdump

naaagt コマンドが送受信する SNMP パケットの Var Bind リストの内容を標準出力に表示します。

-n

naaagt コマンドをデーモン化しません。

### 外部の影響

#### 環境変数

次の環境変数を SnmpNaa ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合は、マスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は、`/etc/services` ファイルの `snmp` 行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

# trapdestagt

---

## 形式

```
trapdestagt [ -aperror ] [ -apwarn ] [ -aptrace ] [ -apconfig ]  
            [ -appacket ] [ -aptrap ] [ -apaccess ] [ -apemanate ]  
            [ -apverbose ] [ -apuser ] [ -apall ] [ -help ]  
            [ -retry N ]
```

## 機能

HP 企業固有 MIB の Trap グループ ( enterprises.hp.nm.snmp.trap ) を提供します。

## 格納場所

- Solaris , HP-UX (IPF) の場合 : /opt/CM2/ESA/bin
- AIX , Linux , Linux (IPF) の場合 : /usr/sbin

## 引数

-aperror

エラーログを出力します。

-apwarn

エラーや警告ログを出力します。

-aptrace

トレースログを出力します。

-apconfig

構成ファイルに関するログを出力します。

-appacket

パケットの組み立て、および解析ログを出力します。

-aptrap

トラップメッセージに関するログを出力します。

-apaccess

エージェント処理に関するログを出力します。

-apemanate

マスターエージェントとサブエージェントに関するログを出力します。

## 5. コマンドとプロセス

### trapdestagt

-apverbose

冗長ログを出力します。

-apuser

ユーザーログを出力します。

-apall

すべてのログを出力します。

-help

コマンドシンタクスを表示します。

-retry N

マスターエージェントの接続を、N 秒単位で実行します。

### 外部の影響

#### 環境変数

次の環境変数を SnmpTrpDst ファイルで指定します。

#### SR\_SNMP\_TEST\_PORT

マスターエージェントの SNMP 受信ポートを設定します。サブエージェントでこの環境変数が指定された場合は、マスターエージェントと接続する際のデータとして使用します。そのため、マスターエージェントに指定したポート番号と同じ値にしなければなりません。この環境変数が指定されていない場合は、`/etc/services` ファイルの `snmp` 行の値を使用します。通常は指定する必要はありません。マスターエージェントの SNMP 受信ポートを変更する場合には指定する必要があります。

# 6

## トラブルシューティング

この章では、SNMP エージェントでトラブルの原因になりやすい項目やトラブルへの対処方法などについて説明します。

- 
- 6.1 対処の手順
  - 6.2 問題の把握
  - 6.3 ログの取得
  - 6.4 資料の採取
  - 6.5 トラブルへの対処方法
  - 6.6 ログ情報の採取方法
-

## 6.1 対処の手順

---

SNMP エージェントの動作でトラブルが起きた場合の対処の手順を次に示します。

1. 問題の把握

問題の兆候から問題の状況を明確にします。問題の状況の把握については、「6.2 問題の把握」を参照してください。

2. ログおよび資料の採取

SNMP エージェントはログを常時出力しています。ログの出力内容については、「6.3 ログの取得」を参照してください。

また、トラブルの対処に必要な資料を採取し、システム管理者へ連絡してください。

資料の採取については、「6.4 資料の採取」を参照してください。

3. トラブルへの対処方法の確認

SNMP は UDP が基になっています。ただし、UDP にはエラーチェックがなく、メッセージ受領の保証もありません。UDP を使用しているために、マネージャーと SNMP エージェント間の通信で問題が発生することがあります。SNMP エージェントのトラブルを解決するときは、注意してください。

「6.5 トラブルへの対処方法」を参照し、トラブルが発生している部分、または問題の範囲を切り分け、対処に従ってください。

また、SNMP エージェントのトラブルを解決するときは、次に示す説明も参照してください。

- 「1. SNMP エージェントの概要」
- 「4. MIB オブジェクト一覧」

SNMP エージェントや `/etc/SnmpAgent.d/snmpd.extend` ファイルの問題でもない場合は、OS のマニュアルを参照してください。

なお、この章で記載しているすべての実行ファイル名については、OS によってパス名が異なるため、ご利用になる OS に対して「付録 A SNMP エージェントのファイルの一覧」を参照してください。

## 6.2 問題の把握

---

問題発生の兆候から問題の状況を明確にします。問題発生の兆候を発見したときは、次に示す基本的な情報を収集します。

### 問題の影響範囲

発生した問題の影響範囲を確認します。影響範囲は次の内容を確認する必要があります。

- エージェントの問題かマネージャーの問題かの区別
- エージェントの問題か `/etc/SnmpAgent.d/snmpd.extend` ファイルの問題かの区別

エージェントの問題かマネージャーの問題かを区別するのは、エージェントでの問題がマネージャーの問題のように見えることがあるためです。マネージャーが SNMP によってデータを送受信するとき、通常はエージェント側に問題があります。例えば、マネージャーが持っているネットワーク上のあるノードに関する情報が誤っている場合、エージェントが誤った情報を送信していることが考えられます。

これらの内容については、「6.5 トラブルへの対処方法」を参照してください。

### SNMP エージェントの影響部分

SNMP エージェントのどの部分に影響があったか確認します。すべての操作に影響があったか、あるいは幾つかの操作だけに影響があったかを確認します。

### 問題の状況

ネットワークの構成（ハードウェア、ソフトウェア、ファイル、セキュリティ、ユティリティなど）で、何が変わったかを確認します。

### 問題の継続時間

問題が連続（毎回）か断続的（時々）かを確認します。

### 問題が発生したときの動作

問題が発生したとき、インスタンスに何が発生したかを確認します。確認する内容を次に示します。

- どの操作が選択されたのか。
- どのコマンドが実行されたのか。
- どのようなデータが要求または送信されたのか。

## 6.3 ログの取得

ログは異常が発生した場合に、その原因を調査するためにシステム管理者が使用します。拡張 MIB 定義機能を使用している場合は MIB 取得時に実行しているコマンドのトレースを出力して問題個所の特定に使用します。

マスターエージェントおよびサブエージェントは、デフォルトで次の内容でログを常時出力します。

- ログの種類：警告ログおよびエラーログ
- ログの出力先：/var/adm/snmpd.logn (n：ログファイルの面を示す値 (1 ~ 10))
- ログのファイルサイズ：10 メガバイト

なお、取得するログの種類、ログの出力先およびログのファイルサイズと面数は、次の方法で変更できます。

### (1) ログの種類

取得するログの種類は、ログマスクによって選択できます。ログマスクは、ログの取得内容の種類を特定するものです。

マスターエージェントとサブエージェントとでは、ログマスクは異なります。マスターエージェントのログマスクを次の表に示します。

表 6-1 マスターエージェントのログマスク

ログマスク (snmpdm -m で指定する値)			意味
文字列	10 進数	16 進数	
-	0	0x	ログの抑止
FACTORY_TRACE	8388608	0x00800000	トレースログの出力
FACTORY_WARN	268435456	0x10000000	警告ログの出力
FACTORY_ERROR	536870912	0x20000000	エラーログの出力

(凡例)

- : 該当なし。

マスターエージェントは、ログマスクの指定を省略すると、ログマスク値にはエラーログと警告ログとを加算した値が設定されます。

マスターエージェントのログを変更するには、現在稼働中のマスターエージェントを kill コマンドで停止させます。そのあと、次に示すように入力します。

```
/usr/sbin/snmpdm -m 8388608
```

複数のログマスクを指定したい場合に、10 進数、16 進数を指定するとき、それぞれの口

グマスク値を加算した値を指定してください。文字列で指定するときは、`-m` に続けて文字列を指定してください。例を次に示します。

```
/usr/sbin/snmpdm -m FACTORY_TRACE FACTORY_WARN FACTORY_ERROR
```

`-m` オプションを文字列で指定する場合で、ほかのオプションと組み合わせて指定するとき、`-m` オプションは最後に指定してください。

サブエージェントのログマスク値を次の表に示します。

表 6-2 サブエージェントのログマスク値

ログマスク (各サブエージェントの コマンドのオプションとして指定)	意味
<code>-aperror</code>	エラーログ
<code>-apwarn</code>	エラー、および警告
<code>-aptrace</code>	トレース
<code>-apconfig</code>	構成ファイル
<code>-appacket</code>	パケットの組み立て、および解析
<code>-aptrap</code>	トラップメッセージ
<code>-apaccess</code>	エージェント処理
<code>-apemanate</code>	マスターエージェント・サブエージェント
<code>-apverbose</code>	冗長ログ
<code>-apuser</code>	ユーザーログ
<code>-apall</code>	すべてのログ

サブエージェントは、ログマスクの指定を省略するとログを取得しません。サブエージェントのログを取得するには、現在稼働中のサブエージェントを `kill` コマンドで停止させます。そのあと、次に示すように入力します。

```
/usr/sbin/extsubagt -aperror
```

複数のログマスクを指定する場合は、指定したいログマスク値を続けて指定します。例を次に示します。

```
/usr/sbin/extsubagt -aperror -apwarn -aptrace
```

## (2) ログの出力先

ログの出力先は `snmpdm` プロセスの環境変数定義ファイル (`SnmpMaster`) の `SR_LOG_DIR` 環境変数で変更できます。なお、ファイル名は `snmpd.logn` で固定です。変更はできません。

## 6. トラブルシューティング

SR\_LOG\_DIR 環境変数については、「5. コマンド」の「snmpdm」を参照してください。

ログの出力先を変更する手順を次に示します。

1. snmpstop コマンドで SNMP エージェントを停止する。
2. SR\_LOG\_DIR 環境変数でログの出力先を指定する。

(例)

```
SR_LOG_DIR=/tmp/esalog
export SR_LOG_DIR
```

3. snmpstart コマンドで SNMP エージェントを再起動する。

また、資料採取コマンドの jp1esalog.sh.def は、デフォルトでは /var/adm/snmpd.logn を収集する設定になっています。jp1esalog.sh.def コマンド実行でのログ出力先を変更したい場合は、vi エディタなどを使用してログの出力先を変更します。jp1esalog.sh.def コマンドの編集例を次に示します。

変更前

```
COLFILE="$COLFILE ./var/adm/snmpd.log* ./var/adm/*agt*.log ./etc/SnmpAgent.d ./etc/srconf/agt"
```

変更後

```
COLFILE="$COLFILE ./tmp/esalog/snmpd.log* ./var/adm/*agt*.log ./etc/SnmpAgent.d ./etc/srconf/agt"
```

注意事項

jp1esalog.sh.def コマンドを編集する前に、jp1esalog.sh.def コマンドのバックアップを必ず取得してください。

### (3) ログファイルの面数とサイズ

ログファイルの面数とサイズは次の環境変数で変更できます。

- snmpdm プロセスの環境変数定義ファイル (SnmpMaster) の SNMP\_HTC\_SNMPD\_LOG\_SIZE 環境変数で、ログファイル snmpd.logn (n: 1 ~ 10) のサイズを指定します。SNMP\_HTC\_SNMPD\_LOG\_SIZE 環境変数については、「5. コマンド」の「snmpdm」を参照してください。
- snmpdm プロセスの環境変数定義ファイル (SnmpMaster) の SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数で、ログファイル snmpd.logn の面数 n を指定します。SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数については、「5. コマンド」の「snmpdm」を参照してください。

ログファイルの面数とサイズを変更する手順を次に示します。

1. snmpstop コマンドで SNMP エージェントを停止する。
2. SNMP\_HTC\_SNMPD\_LOG\_SIZE 環境変数でログファイルのサイズを指定する。

(例)

```
SNMP_HTC_SNMPD_LOG_SIZE=10
export SNMP_HTC_SNMPD_LOG_SIZE
```

3. SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数でログファイルの面数を指定する。

(例)

```
SNMP_HTC_SNMPD_LOG_CNT=10
export SNMP_HTC_SNMPD_LOG_CNT
```

4. snmpstart コマンドで SNMP エージェントを再起動する。

注 手順 2. と手順 3. はどちらを先に操作してもかまいません。

#### (4) 注意事項

- ログは、システム管理者がトラブルシューティングの資料として使用します。ログファイルはラップアラウンドで出力されます。デフォルトで、ファイル 1 面当たり 10 メガバイトで、10 面で作成するため、ログの格納先には 100 メガバイトを格納できる領域が必要です。

## 6.4 資料の採取

---

SNMP エージェントでは、次の資料をデフォルトで常時出力しています。

- マスターエージェントの送受信パケットダンプ (16 進ダンプおよび VarBind リスト)
- ネイティブエージェントアダプターの送受信パケットダンプ (16 進ダンプおよび VarBind リスト)

なお、マスターエージェントの送受信パケットダンプおよびネイティブエージェントアダプターの送受信パケットダンプのデフォルトの内容およびデフォルトの変更方法はログの取得と同じです。「6.3 ログの取得」を参照してください。

また、SNMP エージェントでは、トラブル発生時に次の資料を取得できます。

- マスターエージェントの詳細トレースの取得
- コミュニティ名不正時のログ取得

### 6.4.1 マスターエージェントの送受信パケットダンプの取得

マスターエージェントの送受信パケットダンプは、マネージャーからの SNMP 要求に対する応答がない場合や不正な SNMP メッセージを応答している疑いがある場合、どのような SNMP メッセージが送受信されているかを調べるために使用します。

マスターエージェントの送受信パケットの例を次の図に示します。

図 6-1 マスターエージェントの送受信パケットの例

2009/02/27 13:00:00 snmpdm: Incoming Packet, addr = 10.208.45.232:1526, length = 34: at line xxx in file xxx	2009/02/27 13:00:00に、snmpdmがマネージャ(IPアドレス10.208.45.232, ポート番号1526番)から、長さ34バイトの要求(Incoming Packet)を受信したことを示しています。
2009/02/27 13:00:00 snmpdm: 30 20 02 01 00 04 06 70 75 62 6c 69 63 a1 13 02 at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 02 29 56 02 01 00 02 01 00 30 07 30 05 06 01 2b at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 05 00 at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: org = NULL TYPE/VALUE at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: Outgoing Packet, addr = 10.208.45.232:1526, length = 84: at line xxx in file xxx	2009/02/27 13:00:00に、snmpdmがマネージャ(IPアドレス10.208.45.232, ポート番号1526番)へ、長さ84バイトの応答(Outgoing Packet)を送信したことを示しています。sysDescr. 0の値を送信していることがわかります。
2009/02/27 13:00:00 snmpdm: 30 52 02 01 00 04 06 70 75 62 6c 69 63 a2 45 02 at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 02 29 56 02 01 00 02 01 00 30 39 30 37 06 08 2b at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 06 01 02 01 01 01 00 04 2b 53 75 6e 4f 53 20 6e at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 65 74 6d 73 75 6e 31 31 20 35 2e 39 20 47 65 6e at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 65 72 69 63 5f 31 31 38 35 35 38 2d 32 36 20 73 at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: 75 6e 34 75 at line xxx in file xxx	
2009/02/27 13:00:00 snmpdm: sysDescr. 0 = SunOS netmsun11 5.9 Generic_118558-26 sun4u at line xxx in file xxx	

## 6.4.2 ネイティブエージェントアダプターの送受信パケットダンプの取得

ネイティブエージェントアダプターの送受信パケットダンプは、マネージャからの SNMP 要求に対する応答がない場合や不正な SNMP メッセージを応答している疑いがある場合、標準 MIB オブジェクトの取得・設定に関して、どのような SNMP メッセージを送受信しているのかを調べるために使用します。

ネイティブエージェントアダプター送受信パケットの例を次の図に示します。

図 6-2 ネイティブエージェントアダプター送受信パケットの例

<pre> naaagt: Outgoing Packet, length = 41:                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  30 27 02 01 00 04 06 70 75 62 6c 69 63 a1 1a 02                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  01 02 02 01 00 02 01 00 30 0f 30 0d 06 09 2b 06                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  01 02 01 04 14 01 03 05 00                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt: iso.3.6.1.2.1.4.20.1.3 = NULL TYPE/VALUE                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt: Incoming Packet, length = 50:                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  30 30 02 01 00 04 06 70 75 62 6c 69 63 a2 23 02                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  01 02 02 01 00 02 01 00 30 18 30 16 06 0e 2b 06                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  01 02 01 04 14 01 03 0a 81 50 6c 4c 40 04 ff ff                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt:  ff 00                 at line xxx in file xxx 2009/02/26 20:12:41 naaagt: iso.3.6.1.2.1.4.20.1.3.10.208.108.76 = 255.255.255.0                 at line xxx in file xxx </pre>	<p>2009/02/26 20:12:41に、naaagtがネイティブエージェントへ長さ41バイトの要求(Outgoing Packet)を送信したことを示しています。</p> <p>2009/02/26 20:12:41に、naaagtがネイティブエージェントから長さ50バイトの応答(Incoming Packet)を受信したことを示しています。 iso.3.6.1.2.1.4.20.1.3.10.208.108.76の値を受信していることがわかります。</p>
--	---

**注意事項**

HP-UX (IPF) ではSNMP エージェントがネイティブエージェントなのでネイティブエージェントアダプターは提供していません。そのため、送受信パケットダンプは取得できません。

### 6.4.3 マスターエージェントの詳細トレースの取得

マスターエージェントの詳細トレースには、次に示す内容が出力されます。

- マスターエージェントとマネージャー間で、どのような SNMP の要求や応答があり、どの要求や応答が成功したのか、または失敗したのか
- マスターエージェントとサブエージェント間で、どのような MIB 値の要求や応答があり、どの要求や応答が成功したのか、または失敗したのか

マスターエージェントの詳細トレースに出力された内容を利用すると、何が原因で、マスターエージェントとマネージャーの間の通信、またはマスターエージェントとサブエージェントの間の通信が失敗したのかを調査することができます。マスターエージェ

ントの詳細トレースを取得する方法を次に示します。

1. snmpdm を停止する。

snmpdm プロセスに対して終了シグナルを通知するために次のコマンドを実行します。

```
Kill -15 snmpdmのプロセス番号
```

snmpdm のプロセス番号は次の方法で求めます。  
次のコマンドを実行します。

```
/opt/CM2/ESA/bin/snmpcheck
```

コマンドを実行すると次のように SNMP エージェントのプロセス一覧が表示されます。その表示中の snmpdm 行の pid= の値がプロセス番号です。

```
snmpdm    running pid=11293
mib2agt   running pid=11330
hp_unixagt    running pid=11331
trapdestagt    running pid=11348
extsubagt    running pid=11384
htc_unixagt1   running pid=11366
htc_unixagt2   running pid=11367
```

2. マスターエージェントを起動する。

ログを出力するファイルは任意です。ここでは /tmp/esa.packet.log とします。

```
SR_SNMP_TEST_PORT=SNMPエージェントのポート番号 (通常は161です)
export SR_SNMP_TEST_PORT
/usr/sbin/snmpdm -tcplocal -hexdump -vbdump -apall
```

3. mib2agt および hp\_unixagt を起動する ( HP-UX (IPF) の場合 )

```
/sbin/init.d/SnmpMib2 start
/sbin/init.d/SnmpHpunix start
```

4. snmpdm が起動していることを確認する。

```
/opt/CM2/ESA/bin/snmpcheck
```

Solaris および AIX の場合、詳細トレースを取得中にコマンドを実行した端末からログアウトすると、このコマンドで生成したプロセスに HUP シグナルが送信され、詳細トレースが取得できないときがあります。端末からログアウトしなければ、この現象は発生しません。ログアウトしなければいけない場合は次の方法で実施してください。

詳細トレースコマンド実行後にそのコマンドのプロセス ID を検索してから、次のコマンドを実行してください。

```
/usr/bin/nohup -p -a "プロセスのPID"
```

nohup コマンドの詳細については、man ページなどを参照してください。

## 6.4.4 コミュニティ名不正時のログ取得

マネージャーが通知した SNMP リクエストのコミュニティ名が、SNMP エージェントで定義されているコミュニティ名と一致しない場合、SNMP リクエストの送信元 IP アドレスとコミュニティ名をログに表示することができます。

次に、10.111.98.36 のノードからコミュニティ名が「bad-comm」で送信した SNMP リクエストがコミュニティ不正である場合の出力例を示します。

注

コミュニティ名が 255 文字より大きい場合は 255 文字までを表示します。

```
2003/01/09 19:05:11
snmpdm: Authentication failure, bad community string
Message from 10.111.98.36:2869, community = bad-comm.
```

次にコミュニティ名不正時の SNMP リクエストの送信元 IP アドレスおよびコミュニティ名取得の設定方法を示します。この作業はすべてスーパーユーザーで実行してください。なお、SNMP エージェントはデフォルトの設定ではこのログを取得しません。

1. 次に示すファイルを vi などのエディタを使用して編集する。

```
/etc/rc.config.d/SnmpMaster (Solarisの場合)
/usr/CM2/ESA/opt/SnmpMaster (AIXの場合)
/opt/CM2/ESA/opt/SnmpMaster (HP-UX (IPF), Linux, Linux (IPF) の場合)
```

2. 次の行をファイルの最終行に追加する。

```
SNMP_HTC_AUTH_LOG=1
export SNMP_HTC_AUTH_LOG
```

3. ファイルを保存する。
4. SNMP エージェントを再起動する。

次のコマンドを実行します。

```
/opt/CM2/ESA/bin/snmpstart (AIX以外の場合)
/usr/CM2/ESA/bin/snmpstart (AIXの場合)
```

### 注意事項

snmpstart コマンドは SNMP エージェントをいったん停止したあとに SNMP エージェントを起動します。

## 6.5 トラブルへの対処方法

---

この節では、次に示す問題について対処方法を説明します。

- SNMP エージェント起動時の問題
- SNMP エージェント運用上の問題

なお、SNMP エージェントを正常に動作させるために、通常の使用時は次に示すことに注意してください。

- SNMP エージェントが、ハードウェア、ソフトウェアの必要な条件を満たすように設定してください。
- オリジナルファイルのバックアップを取得しないで、`/etc/SnmpAgent.d/snmpd.conf` や `/etc/SnmpAgent.d/snmpd.extend` のような SNMP エージェントのファイルを変更しないでください。これらのファイルはバックアップを取得しておき、通常はバックアップを使用します。ファイルに問題が発生した場合、オリジナルファイルを使用すれば通常の設定値に回復できます。オリジナルファイルは、`/opt/OV/newconfig/EAGENT-RUN` ディレクトリにあります。

### 6.5.1 SNMP エージェント起動時の問題

SNMP エージェントが起動できない場合、次の要因が考えられます。

- ソフトウェアバージョンおよびファイルのアクセス権を確認します。  
SNMP エージェントのファイルのアクセス権のデフォルトは、マスターエージェントやサブエージェントでは `root` だけが実行できます。
- SNMP 受信ポート番号を確認します。  
SNMP エージェントの SNMP 受信ポート番号がネイティブエージェントや他社 SNMP エージェントの SNMP 受信ポート番号と重複している場合、「3.2 SNMP エージェントの SNMP 受信ポートの変更」を参照して、SNMP エージェントの SNMP 受信ポート番号を変更してください。
- 環境変数定義ファイルを確認します。  
環境変数定義ファイルを `Snmp` で始まるファイル名でバックアップしている場合、バックアップファイルが環境変数定義ファイルとして読み込まれていることがあります。環境変数定義ファイルのバックアップの注意事項については、「3.5 運用上の注意事項」を参照してください。

### 6.5.2 SNMP エージェントの運用上の問題

SNMP エージェントを運用する上での次の問題について対処方法を説明します。

- MIB が取得できない。
- SNMP トラップがマネージャーに届かない。

## 6. トラブルシューティング

- SNMP エージェントの拡張機能が使用できない。

### (1) MIB が取得できない場合

SNMP エージェントが起動しているにもかかわらず、マネージャー製品から MIB 値が取得できない場合、次の要因が考えられます。

- SNMP エージェントとマネージャーが通信できるか確認します。ネットワーク構成に問題があることも考えられます。ネットワークの接続を確認するには ping コマンドを実行します。
- NNM および SubManager の snmpget コマンドによって SNMP エージェントの MIB 値を調査します。
- 調査のために SNMP エージェントの MIB の一部またはすべてのダンプを取得するには、NNM および SubManager の snmpwalk コマンドを使用します。
- SNMP エージェントに設定されたオブジェクト識別子が、マネージャーに設定されたオブジェクト識別子と一致しているかどうかを確認します。
- SNMP SetRequest を実行しようとしている場合は、SNMP エージェントが SNMP SetRequest に応答するように設定されているかどうかを確認します。デフォルトでは、マネージャーは SNMP エージェントの MIB 値を変更できません。SNMP エージェントを SNMP SetRequest に応答するように設定するには、SNMP エージェントの /etc/SnmpAgent.d/snmpd.conf ファイルに set コミュニティ名を追加します。
- 情報の検索が正しいかを確認するには、NNM の [ MIB のブラウズ : SNMP... ] の操作を使用します。
- Solaris , AIX , Linux , Linux(IPF) でネイティブエージェント提供の MIB 値が取得できない場合、SNMP エージェントのネイティブエージェントアダプターと OS が提供するネイティブエージェント間のコミュニティ名が一致しているか確認してください。  
なお、ネイティブエージェント提供の MIB 値とは、JP1/Cm2/SSO のリソースでは、ネットワークサマリ、回線利用率、インタフェーストラフィック、IP トラフィック、ICMP トラフィック、TCP トラフィック、および UDP トラフィックです。
- SNMP エージェントで提供するすべての MIB 値が取得不可となると、次の手順を実行してください。
  1. マネージャー製品と SNMP エージェントのコミュニティ名が一致しているかを確認してください。  
SNMP エージェントのコミュニティ名を変更したあとに、SNMP エージェントまたは OS を再起動しているか確認してください。
  2. マネージャー製品と SNMP エージェントの間にファイアウォールがあり、SNMP 通信が許可されていることを確認してください。  
ファイアウォールの設定を見直してください。  
また、OS が提供するパケットトレースコマンドなどでパケットトレースを取得して、OS 側まで SNMP リクエストが届いているか確認してください。  
使用方法については man ページなどで確認してください。

AIX の場合 : iptrace コマンド  
 Solaris の場合 : snoop コマンド  
 Linux, Linux(IPF) の場合 : tcpdump コマンド  
 HP-UX(IPF) の場合 : nettl コマンド

- MIB 値がタイムアウトや noSuchName エラーで取得できないことがあるとき  
 マネージャー製品からの SNMP リクエストのタイムアウト時間が短い場合に、このような現象が発生します。  
 「3.5 運用上の注意事項」を参照してタイムアウト時間が適切が見直してください。
- そのほかの MIB 値が取得不可となるとき  
 Solaris, AIX, Linux, Linux (IPF) で SNMP エージェントが MIB 値を取得するために使用するコマンドがインストールされていません。  
 /opt/CM2/ESA/bin/snmpcmdchk コマンドを実行して、必要なコマンドがインストールされているか確認してください。詳細については「2.2.1(2) MIB 値の取得で使用するコマンドのインストール (HP-UX (IPF) 以外の場合)」を参照してください。

## (2) SNMP トラップがマネージャーに届かない場合

SNMP エージェントが起動しているにもかかわらず、SNMP トラップがマネージャーに届かない場合、次の要因が考えられます。なお、マネージャーシステムと SNMP エージェントシステム間の問題については、「(1) MIB が取得できない場合」を参照してください。

- SNMP エージェントのトラップのあて先が正しく設定されているかどうかを確認します。トラップのあて先については、「2.4.3 トラップのあて先の設定」を参照してください。
- NNM および SubManager の SNMP コマンドを使用して、リモート SNMP ノードでの SNMP オペレーションを確認します。

## (3) SNMP エージェントの拡張機能が使用できない場合

SNMP エージェントが起動しているにもかかわらず、SNMP エージェントの拡張機能が使用できない場合、次の要因が考えられます。なお、マネージャーシステムと SNMP エージェントシステム間の問題については、「(1) MIB が取得できない場合」を参照してください。

/etc/SnmpAgent.d/snmpd.extend ファイルの問題

/etc/SnmpAgent.d/snmpd.extend ファイルのトラブルシューティングは、最初は部分的に実行して、次にネットワーク全体について実行します。/etc/SnmpAgent.d/snmpd.extend ファイルのトラブルを解決するには、次に示す方法を使用します。

- コマンドの応答が正しいかどうかを確認するには、OS コマンドラインから /etc/SnmpAgent.d/snmpd.extend ファイル中の各コマンドを実行します。

## 6. トラブルシューティング

- 次に示すコマンドを使用して、コマンドが正確に実行されたか確認します。

```
echo $?
```

- コマンドが引数を持つ場合は、引数を確認します。このために独自に必要なパラメータをすべて設定し、コマンドを実行します。  
シンタクスエラーは、extsubagt 起動時に標準エラーに出力されます。
- SNMP エージェントは /etc/SnmpAgent.d/snmpd.extend ファイルの読み込み中にエラーを検知すると、エラーが発生した行および正しいシンタクスを表示します。
- /etc/SnmpAgent.d/snmpd.extend ファイルに定義されたコマンドが実行できるかどうか調査します。
- コマンドに実行認可が設定されているかどうかを確認します。
- 情報を取得するために使用するオブジェクト識別子が、正しいコマンドを実行するかどうかを調査します。そのオブジェクトで何のコマンドを実行するかを確認するには、ログマスクを次のように設定します。

```
/usr/sbin/extsubagt -apall
```

- 実行しているコマンドの出力が正しいデータ形式と一致しているかどうかを確認します。
- /etc/SnmpAgent.d/snmpd.extend ファイル中のコマンドが正しいかをどうか確認します。例えば、/usr を /user と指定したことなどが考えられます。

### マネージャーからの問題

エージェントでの問題を解決したあとでも、まだ問題がある場合は、次に示す内容を調査します。

- マネージャーから /etc/SnmpAgent.d/snmpd.extend ファイルの各オブジェクトに SNMP 要求をし、ファイルが正しく動作するかを確認します。
- SNMP SetRequest をしたあと、SNMP GetRequest によって値が正しく設定されたかどうかを確認します。

## 6.6 ログ情報の採取方法

---

SNMP エージェントでは、トラブルが発生したときに迅速に資料を採取できるように、資料採取ツールを提供しています。「5. コマンドとプロセス」の「jp1esalog.sh.def」を参照してください。



# 付録

---

付録 A SNMP エージェントのファイルの一覧

---

付録 B ポート番号一覧

---

付録 C カーネルパラメーター一覧

---

付録 D SNMP エージェントの前提とするパッチ，プロセス（サービス）一覧

---

付録 E IBM RISC システム /6000 SP2(RS/6000 SP2) へのインストールとアンインストール

---

付録 F 各バージョンの変更内容

---

付録 G (3020-3-T04-01) の変更内容

---

付録 H 用語解説

---

## 付録 A SNMP エージェントのファイルの一覧

OS ごとに、SNMP エージェントのファイルの一覧を示します。

### 付録 A.1 SNMP エージェントのファイルの一覧 (Solaris の場合)

種類	パス名	ファイル名	
ロードモジュール	/usr/sbin (シンボリックリンク)	extsubagt	
		/opt/CM2/ESA/bin	snmpdm
		extsubagt	
		hp_unixagt	
		htc_unixagt1	
		naaagt	
		trapdestagt	
		htc_unixagt3	
		htc_monagt1	
		/opt/OV/bin	snmptrap
			systemtrap
開始・終了コマンド	/opt/CM2/ESA/bin	snmpstart	
		snmpstop	
		snmpcheck	
資料採取コマンド	/opt/CM2/ESA/bin	jp1esalog.sh.def	
MIB 値取得コマンド	/opt/CM2/ESA/bin	cpuutil.exe	
		freememory.exe	
		physmemory.exe	
		swapconfig.exe	
		diskBusy.exe	
		cpuInfo.exe	
		processorCpuTime.exe	
		diskTime.exe	
MIB 値作成ディレクトリ	/opt/CM2/ESA/out	cpuutil.out	
		freememory.out	
		physmemory.out	
		swapconfig.out	

種類	パス名	ファイル名
		diskBusy.out
		cpuInfo.out
		processorCpuTime.out
		diskTime.out
		temp_processorCpuTime.out
		mpstat.out
		mpstat.err.tmp
		mpstat.err
		mpstat.all
マスターエージェント / サブエージェント間通信	/tmp/.AgentSockets	配下のファイル
パッチファイル履歴	/opt/CM2/ESA	esa_spackinfo
ソケット通信	/opt/CM2/ESA/sockets	agt3_mon1
		mon1_comm
構成定義ファイル	/etc/SnmpAgent.d	snmpd.conf
	/etc	snmpd.conf (シンボリックリンク)
	/etc/srconf/agt	naa.cnf
		snmpd.cnf
		snmpd.cnf~
		snmpd.jnk
	/etc/opt/OV/share/conf	opConfCharCode.conf
		snmpmib
		snmpmib.bin
	/opt/CM2/ESA/ext	配下のファイル
拡張 MIB 定義サンプル	/opt/OV/prg_samples/eagent	snmpd.extend
		change_num_widgets
		get_processes
		list_processes
		memory.curly
		memory.larry
		memory.moe
		memory.public
		num_widgets
		root_processes

種類	パス名	ファイル名
		update_inetd
		user_disk_space
MIB 定義文	/var/opt/OV/share/snmp_mibs/ eagent	rfc1213-MIB-II
		hp-unix
		hitachi-cometAgt
		hitachi-cometAgt-aix
		hitachi-cometAgt-solaris
		hitachi-cometAgt-linux
		hitachi-cometAgt-tru64
起動 / 停止ファイル	/sbin/init.d	esa
	/opt/CM2/ESA/init.d	SnmpMaster
		SnmpHpunix
		SnmpTrpDst
		SnmpHtcunix1
		SnmpExtSubagent
		SnmpNaa
		SnmpHtcunix3
		SnmpHtcmonagt1
停止ファイル (シンボリックリンク)	/etc/rc0.d	K02esa
	/etc/rc1.d	K02esa
起動ファイル (シンボリックリンク)	/etc/rc2.d	S97esa
環境変数定義ファイル	/etc/rc.config.d	SnmpMaster
		SnmpHpunix
		SnmpTrpDst
		SnmpNaa
		SnmpHtcmonagt1
		SnmpHtcunix1
		SnmpHtcunix3
		SnmpExtAgt
インストール情報	/etc/.hitachi/before	before_5700
	/etc/.hitachi/after	after_5700
	/etc/.hitachi/remove	remove_5700
インストールログ	/tmp	esa.log

種類	パス名	ファイル名
インストール用 バックアップファイル	/opt/OV/newconfig/EAGENT-RUN	snmpdm
		hp_unixagt
		trapdestagt
		extsubagt
		naaagt
		htc_unixagt1
		snmptrap
		snmpmib
		snmpmib.bin
		systemtrap
		snmpd.conf
		esafsys.conf
		snmpd.cnf
		naa.cnf
		SnmExtSubagent
		SnmHtcunix1
		config_new_SnmMaster
		config_new_SnmHpunix
		config_new_SnmTrpDst
		config_new_SnmNaa
		rc.config
		init_new_SnmMaster
		init_new_SnmHpunix
		init_new_SnmTrpDst
		init_new_SnmNaa
		htc_unixagt3
		htc_monagt1
		SnmHtcunix3
		SnmHtcmonagt1
		INIT.SNMPDX ( Solaris9 の場合 )
		SNMPDX.REG ( Solaris9 の場合 )
		snmpdx.reg.tmp ( Solaris9 の場合 )

種類	パス名	ファイル名
		snmpdx.reg.tmp1 (Solaris9 の場合)
		INIT.SMA (Solaris10 の場合)
		init.sma.tmp (Solaris10 の場合)
		SVC-SMA (SMF 対応した Solaris10 の場合)
		svc-sma.tmp (SMF 対応した Solaris10 の場合)
	/opt/OV/newconfig/EAGENT-RUN/rc.config.d/	SnmpExtAgt
		SnmpHtcmonagt1
		SnmpHtcunix1
		SnmpHtcunix3
ログファイル	/var/adm/	snmpd.log <sub>n</sub>
	/var/opt/CM2/ESA/log	htc_monagt1.log
		htc_monagt1.log.old
シンボリックリンクファイル	/usr/OV/bin	snmptrap
		systemtrap
	/usr/OV/conf	snmpmib.bin
ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf
ファイルシステム定義ファイル (エラー)	/etc/SnmpAgent.d	esafilesys.conf.err
OS コマンドインストール確認	/opt/CM2/ESA/bin	snmpcmdchk
動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf

注 n の値は 1 ~ SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数で指定した値。  
SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数を指定していない場合は最大値の 10。

## 付録 A.2 SNMP エージェントのファイルの一覧 (AIX の場合)

種類	パス名	ファイル名
ロードモジュール	/usr/sbin	snmpdm
		extsubagt
		hp_unixagt

種類	パス名	ファイル名
		htc_unixagt1
		naaagt
		trapdestagt
		htc_unixagt3
		htc_monagt1
	/usr/OV/bin	snmptrap
	systemtrap	
開始・終了コマンド	/usr/CM2/ESA/bin	snmpstart
		snmpstop
		snmpcheck
資料採取コマンド	/usr/CM2/ESA/bin	jp1esalog.sh.def
MIB 値取得コマンド	/usr/CM2/ESA/bin	disk.exe
		freememory.exe
		loadave.exe
		page.exe
		phymemory.exe
		process.exe
		reconfigure.exe
		swapconfig.exe
		systemactive.exe
		usingmemory.exe
		vmactive.exe
		vmforks.exe
		vmtotal.exe
		cpuInfo.exe
		processorCpuTime.exe
		diskTime.exe
MIB 値作成ディレクトリ	/usr/CM2/ESA/out	freememory.out
		loadave.out
		pageSize.out
		phymemory.out
		process.out
		swapconfig.out
		systemactive.out
		vmtotal.out

種類	パス名	ファイル名
		cpuInfo.out
		vmactive64.out
		vmtotal64.out
		vmforks64.out
		processorCpuTime.out
		diskTime.out
		page.out
		disk.out
		vmactive.out
		vmforks.out
		h_process.out
		htc_swapconfig.out
		temp_processorCpuTime.out
		temp_vmtotal.out
		mpstat.out
		mpstat.err.tmp
		mpstat.err
		mpstat.all
マスターエージェント / サブエージェント間通信	/tmp/.AgentSockets	配下のファイル
パッチファイル履歴	/usr/CM2/ESA	esa_spackinfo
ソケット通信	/usr/CM2/ESA/sockets	agt3_mon1
		mon1_comm
構成定義ファイル	/etc/SnmpAgent.d	snmpd.conf
	/etc/srconf/agt	snmpd.cnf
		naa.cnf
		snmpd.cnf~
		snmpd.jnk
		snmpd.jnk
	/usr/OV/conf	opConfCharCode.conf
		snmpmib
		snmpmib.bin
	/usr/CM2/ESA/ext	配下のファイル
拡張 MIB 定義サンプル	/usr/OV/prg_samples/eagent	snmpd.extend
		change_num_widgets

種類	パス名	ファイル名
		get_processes
		list_processes
		memory.curly
		memory.larry
		memory.moe
		memory.public
		num_widgets
		root_processes
		update_inetd
		user_disk_space
MIB 定義文	/usr/OV/snmp_mibs/eagent	rfc1213-MIB-II
		hp-unix
		hitachi-cometAgt
		hitachi-cometAgt-aix
		hitachi-cometAgt-solaris
		hitachi-cometAgt-linux
		hitachi-cometAgt-tru64
起動ファイル	/etc/inittab に登録	-
	/usr/CM2/ESA/bin	esa
停止ファイル	/etc/rc.shutdown	-
環境変数定義ファイル	/usr/CM2/ESA/opt	SnmpMaster
		SnmpHpunix
		SnmpTrpDst
		SnmpNaa
		SnmpNative
		SnmpHtcmonagt1
		SnmpHtcunix1
		SnmpHtcunix3
		SnmpExtAgt
インストール情報	/etc/.hitachi/before	before_112B
	/etc/.hitachi/after	after_112B
	/etc/.hitachi/remove	remove_112B
インストールログ	/tmp	esa.log
インストール用 バックアップファイル	/usr/OV/newconfig/EAGENT-RUN	snmpdm
		hp_unixagt

種類	パス名	ファイル名
		trapdestagt
		extsubagt
		naaagt
		htc_unixagt1
		htc_unixagt3
		htc_monagt1
		snmptrap
		snmpmib
		snmpmib.bin
		systemtrap
		snmpd.conf
		esafilesys.conf
		snmpd.cnf
		naa.cnf
	/usr/OV/newconfig/EAGENT-RUN/ rc.config.d	SnmpExtAgt
		SnmpMaster
		SnmpMib2
		SnmpHpunix
		SnmpTrpDst
		SnmpNaa
SnmpHtcunix1		
SnmpHtcunix3		
SnmpHtcumonagt1		
SnmpNative		
ログファイル	/usr/adm/	snmpd.logn
	/usr/CM2/ESA/log	htc_monagt1.log
		htc_monagt1.log.old
ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf
ファイルシステム定義ファイル (エラー)	/etc/SnmpAgent.d	esafilesys.conf.err
OS コマンドインストール確認	/usr/CM2/ESA/bin	snmpcmdchk

(凡例) - : 該当なし。

注 n の値は 1 ~ SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数で指定した値。

SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数を指定していない場合は最大値の 10。

### 付録 A.3 SNMP エージェントのファイルの一覧 (Linux の場合)

種類	パス名	ファイル名
ロードモジュール	/usr/sbin	snmpdm
		extsubagt
		hp_unixagt
		htc_unixagt1
		naaagt
		trapdestagt
		htc_unixagt3
		htc_monagt1
		/opt/OV/bin
systemtrap		
開始・終了コマンド	/opt/CM2/ESA/bin	snmpstart
		snmpstop
		snmpcheck
資料採取コマンド	/opt/CM2/ESA/bin	jp1esalog.sh.def
MIB 値取得コマンド	/opt/CM2/ESA/bin	linuxSystem.exe
		loadave.exe
		process.exe
		cpuInfo.exe
		linuxPhysMem.exe
		linuxSwap.exe
MIB 値作成ディレクトリ	/opt/CM2/ESA/out	linuxSystem.out
		loadave.out
		process.out
		cpuInfo.out
		linuxPhysMem.out
		linuxSwap.out
		mpstat.out
		mpstat.err.tmp
		mpstat.err
		mpstat.all

種類	パス名	ファイル名	
マスターエージェント / サブエージェント間通信	/tmp/.AgentSockets	配下のファイル	
パッチファイル履歴	/opt/CM2/ESA	esa_spackinfo	
構成定義ファイル	/etc/SnmpAgent.d	snmpd.conf	
		/etc	snmpd.conf (シンボリックリンク)
		/etc/srconf/agt	naa.cnf
			snmpd.cnf
	snmpd.cnf~		
	/etc/opt/OV/share/conf	opConfCharCode.conf	
		snmpmib	
		snmpmib.bin	
	/opt/CM2/ESA/ext	配下のファイル	
	拡張 MIB 定義サンプル	/opt/OV/prg_samples/eagent	snmpd.extend
change_num_widgets			
get_processes			
list_processes			
memory.curly			
memory.larry			
memory.moe			
memory.public			
num_widgets			
root_processes			
update_inetd			
user_disk_space			
MIB 定義文			/var/opt/OV/share/snmp_mibs/eagent
	hp-unix		
	hitachi-cometAgt		
	hitachi-cometAgt-aix		
	hitachi-cometAgt-solaris		
	hitachi-cometAgt-linux		
	hitachi-cometAgt-tru64		
ソケット通信	/opt/CM2/ESA/sockets	agt_mon1	

種類	パス名	ファイル名
		mon1_comm
起動 / 停止ファイル	/etc/rc.d/init.d	esa
停止ファイル (シンボリックリンク)	/etc/rc.d/rc0.d	K65esa
	/etc/rc.d/rc2.d	K65esa
	/etc/rc.d/rc3.d	K65esa
	/etc/rc.d/rc5.d	K65esa
	/etc/rc.d/rc6.d	K65esa
起動ファイル (シンボリックリンク)	/etc/rc.d/rc2.d	S55esa
	/etc/rc.d/rc3.d	S55esa
	/etc/rc.d/rc5.d	S55esa
環境変数定義ファイル	/opt/CM2/ESA/opt	SnmpMaster
		SnmpHpunix
		SnmpTrpDst
		SnmpNaa
		SnmpHtcmonagt1
		SnmpHtcunix1
		SnmpHtcunix3
		SnmpExtAgt
インストール情報	/etc/.hitachi/before	before_112B
	/etc/.hitachi/after	after_112B
	/etc/.hitachi/remove	remove_112B
インストールログ	/tmp	esa.log
インストール用 バックアップファイル	/opt/OV/newconfig/EAGENT-RUN	snmpdm
		hp_unixagt
		trapdestagt
		extsubagt
		naaagt
		htc_unixagt1
		snmptrap
		snmpmib
		snmpmib.bin
		systemtrap
		snmpd.conf
		esafilesys.conf
		snmpd.cnf

種類	パス名	ファイル名	
		naa.cnf	
		esa	
		htc_monagt1	
		htc_unixagt3	
		/opt/OV/newconfig/EAGENT-RUN/rc.config.d	SnmpExtAgt
			SnmpMaster
			SnmpMib2
			SnmpHpunix
			SnmpTrpDst
			SnmpNaa
			SnmpHtcunix1
			SnmpHtcunix3
			SnmpHtcmonagt1
ログファイル	/var/adm/	snmpd.logn	
	/var/opt/CM2/ESA/log	htc_monagt1.log	
		htc_monagt1.log.old	
ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf	
ファイルシステム定義ファイル(エラー)	/etc/SnmpAgent.d	esafilesys.conf.err	
ファイルシステム情報格納ファイル	/opt/CM2/ESA/conf	fileSystemID.db	
OS コマンドインストール確認	/opt/CM2/ESA/bin	snmpcmdchk	
動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf	

注 n の値は 1 ~ SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数で指定した値。  
SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数を指定していない場合は最大値の 10。

## 付録 A.4 SNMP エージェントのファイルの一覧 (HP-UX (IPF) の場合)

種類	パス名	ファイル名
ロードモジュール	/opt/CM2/ESA/bin	snmpdm
		extsubagt
		htc_unixagt1

種類	パス名	ファイル名	
		htc_unixagt2	
		trapdestagt	
	/opt/OV/bin	snmptrap	
		systemtrap	
開始・終了コマンド	/opt/CM2/ESA/bin	snmpstart	
		snmpstop	
		snmpcheck	
資料採取コマンド	/opt/CM2/ESA/bin	jplesalog.sh.def	
マスターエージェント / サブエージェント間通信	/tmp/.AgentSockets	配下のファイル	
パッチファイル履歴	/opt/CM2/ESA	esa_spackinfo	
構成定義ファイル	/etc/SnmpAgent.d	snmpd.conf	
		esafilesys.conf	
	/etc/srconf/agt	snmpd.cnf	
		snmpd.cnf~	
		snmpd.jnk	
	/etc/opt/OV/share/conf	opConfCharCode.conf	
		snmpmib	
		snmpmib.bin	
	拡張 MIB 定義サンプル	/opt/OV/prg_samples/eagent	snmpd.extend
			change_num_widgets
			get_processes
			list_processes
memory.curly			
memory.larry			
memory.moe			
memory.public			
num_widgets			
root_processes			
update_inetd			
user_disk_space			
MIB 定義文			/opt/CM2/ESA/snmp_mibs
	hp-unix		
	hitachi-cometAgt		

種類	パス名	ファイル名
		hitachi-cometAgt-aix
		hitachi-cometAgt-solaris
		hitachi-cometAgt-linux
		hitachi-cometAgt-tru64
起動 / 停止ファイル	/sbin/init.d	esa
停止ファイル (シンボリックリンク)	/sbin/rc1.d	K440esa
起動ファイル (シンボリックリンク)	/sbin/rc2.d	S560esa
環境変数定義ファイル	/opt/CM2/ESA/opt	SnmptMaster
		SnmptMib2
		SnmptHpunix
		SnmptTrpDst
		SnmptExtAgt
		SnmptHtcunix1
		SnmptHtcunix2
インストール情報	/etc/.hitachi/before	before_G12B
	/etc/.hitachi/after	after_G12B
	/etc/.hitachi/remove	remove_G12B
インストールログ	/tmp	esa.log
インストール用バックアップファイル	/opt/CM2/ESA/newconfig	SNMPD.CNF
		esa
		init_new_SnmptHpunix
		init_new_SnmptMib2
		oracle_new_snmpd
		esafilesys.conf
		snmpd.cnf
		snmpd.conf
		snmpmib
		snmpmib.bin
		snmptrap
		systemtrap
	/opt/CM2/ESA/newconfig/rc.config.d	SnmptMaster
		SnmptMib2

種類	パス名	ファイル名
		SnmpHpunix
		SnmpTrpDst
		SnmpExtAgt
		SnmpHtcunix1
		SnmpHtcunix2
ログファイル	/var/adm	snmpd.logn
新規インストールチェックファイル	/opt/CM2/ESA/bin	INSTALLED

注 n の値は 1 ~ SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数で指定した値。  
SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数を指定していない場合は最大値の 10。

## 付録 A.5 SNMP エージェントのファイルの一覧 (Linux (IPF) の場合)

種類	パス名	ファイル名
ロードモジュール	/usr/sbin	snmpdm
		extsubagt
		hp_unixagt
		htc_unixagt1
		naaagt
		trapdestagt
		htc_unixagt3
		htc_monagt1
		/opt/OV/bin
		systemtrap
開始・終了コマンド	/opt/CM2/ESA/bin	snmpstart
		snmpstop
		snmpcheck
資料採取コマンド	/opt/CM2/ESA/bin	jp1esalog.sh.def
MIB 値取得コマンド	/opt/CM2/ESA/bin	linuxSystem.exe
		loadave.exe
		linuxPhysMem.exe
		process.exe
		linuxSwap.exe

種類	パス名	ファイル名	
		cpuInfo.exe	
MIB 値作成ディレクトリ	/opt/CM2/ESA/out	linuxSystem.out	
		loadave.out	
		linuxPhysMem.out	
		process.out	
		linuxSwap.out	
		cpuInfo.out	
		mpstat.out	
		mpstat.err.tmp	
		mpstat.err	
		mpstat.all	
マスターエージェント / サブ エージェント間通信	/tmp/.AgentSockets	配下のファイル	
パッチファイル履歴	/opt/CM2/ESA	esa_spackinfo	
構成定義ファイル	/etc/SnmpAgent.d	snmpd.conf	
		esafsys.conf	
	/etc	snmpd.conf (シンボリック リンク)	
	/etc/srconf/agt	naa.cnf	
		snmpd.cnf	
		snmpd.cnf~	
		snmpd.jnk	
	/etc/opt/OV/share/conf	opConfCharCode.conf	
		snmpmib	
		snmpmib.bin	
	/opt/CM2/ESA/ext	配下のファイル	
	拡張 MIB 定義サンプル	/opt/OV/prg_samples/eagent	snmpd.extend
			change_num_widgets
get_processes			
list_processes			
memory.curly			
memory.larry			
memory.moe			
memory.public			
num_widgets			

種類	パス名	ファイル名
		root_processes
		update_inetd
		user_disk_space
MIB 定義文	/var/opt/OV/share/snmp_mibs/ eagent	rfc1213-MIB-II
		hp-unix
		hitachi-cometAgt
		hitachi-cometAgt-aix
		hitachi-cometAgt-solaris
		hitachi-cometAgt-linux
		hitachi-cometAgt-tru64
ソケット通信	/opt/CM2/ESA/sockets	agt3_mon1
		mon1_comm
起動 / 停止ファイル	/etc/rc.d/init.d	esa
停止ファイル (シンボリックリンク)	/etc/rc.d/rc0.d	K65esa
	/etc/rc.d/rc2.d	K65esa
	/etc/rc.d/rc3.d	K65esa
	/etc/rc.d/rc5.d	K65esa
	/etc/rc.d/rc6.d	K65esa
起動ファイル (シンボリックリンク)	/etc/rc.d/rc2.d	S55esa
	/etc/rc.d/rc3.d	S55esa
	/etc/rc.d/rc5.d	S55esa
環境変数定義ファイル	/opt/CM2/ESA/opt	SnmpMaster
		SnmpHpunix
		SnmpTrpDst
		SnmpNaa
		SnmpHtcmonagt1
		SnmpHtcunix1
		SnmpHtcunix3
		SnmpExtAgt
インストール情報	/etc/.hitachi/before	before_G12B
	/etc/.hitachi/after	after_G12B
	/etc/.hitachi/remove	remove_G12B
インストールログ	/tmp	esa.log
インストール用	/opt/OV/newconfig/EAGENT-RUN	snmpdm

種類	パス名	ファイル名	
バックアップファイル		hp_unixagt	
		trapdestagt	
		extsubagt	
		naaagt	
		hte_unixagt1	
		snmptrap	
		snmpmib	
		snmpmib.bin	
		systemtrap	
		esafilesys.conf	
		snmpd.conf	
		snmpd.cnf	
		naa.cnf	
		esa	
		hte_managt1	
		hte_unixagt3	
		/opt/OV/newconfig/EAGENT-RUN/rc.config.d	SnmpExtAgt
			SnmpMaster
			SnmpMib2
			SnmpHpunix
		SnmpTrpDst	
		SnmpNaa	
		SnmpHtcunix1	
		SnmpHtcunix3	
		SnmpHtcmonagt1	
ログファイル	/var/adm/	snmpd.logn	
	/var/opt/CM2/ESA/log	hte_monagt1.log	
		hte_monagt1.log.old	
ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf	
ファイルシステム定義ファイル (エラー)	/etc/SnmpAgent.d	esafilesys.conf.err	
ファイルシステム情報格納ファイル	/opt/CM2/ESA/conf	fileSystemID.db	
OS コマンドインストール確認	/opt/CM2/ESA/bin	snmpcmdchk	

注 n の値は 1 ~ SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数で指定した値。  
SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数を指定していない場合は最大値の 10。

## 付録 B ポート番号一覧

SNMP エージェントで使用するポート番号およびファイアウォールの通過方向について説明します。

### 付録 B.1 SNMP エージェントで使用するポート番号

SNMP エージェントで使用するポート番号を次の表に示します。

表 B-1 SNMP エージェントで使用するポート番号 ( HP-UX (IPF) , Solaris , AIX の場合 )

サービス名	ポート	内容
-	161/udp	SNMP リクエスト受信
-	8161/udp	ネイティブエージェントアダプターとネイティブエージェントの通信 ( HP-UX(IPF) は使用しない )
-	7161/tcp	サブエージェントとの通信

( 凡例 ) - : 該当なし。

表 B-2 SNMP エージェントで使用するポート番号 ( Linux , Linux (IPF) の場合 )

サービス名	ポート	内容
-	22161/udp	SNMP リクエスト受信
-	161/udp	ネイティブエージェントアダプターとネイティブエージェントの通信
-	22161/tcp	サブエージェントとの通信

( 凡例 ) - : 該当なし。

### 付録 B.2 ファイアウォールの通過方向

ファイアウォールの通過方向を次の表に示します。

表 B-3 ファイアウォールの通過方向 ( HP-UX (IPF) , Solaris , AIX の場合 )

マネージャーホストのポート番号	通過方向	SNMP エージェントのポート番号
ANY		161/udp
ANY		161/udp
162/udp		ANY

表 B-4 ファイアウォールの通過方向 (Linux, Linux (IPF) の場合)

マネージャーホストのポート番号	通過方向	SNMP エージェントのポート番号
ANY		22161/udp
ANY		22161/udp
162/udp		ANY

## 付録 C カーネルパラメーター一覧

SNMP エージェントの実行に必要なリソースを割り当てるために、OS のカーネルパラメーターを調整します。ここでは、調整が必要なカーネルパラメーターについて、OS ごとに説明します。

なお、この節の表中では、次に示す記号を使用して見積もり式を説明します。

記号	意味
1	/etc/SnmpAgent.d/snmpd.extend を設定している場合は 5、設定していない場合は 0
2	/opt/CM2/ESA/ext/ 下に設定している拡張 MIB 定義ファイル数 * 5
3	/etc/SnmpAgent.d/snmpd.extend を設定している場合は 3、設定していない場合は 0
4	/opt/CM2/ESA/ext/ 下に設定している拡張 MIB 定義ファイル数 * 3
5	/opt/CM2/ESA/ext/ 下に設定している拡張 MIB 定義ファイル数

### (1) HP-UX (IPF) の場合

システムリソース	パラメーター	見積もり
ファイルシステム	nfile	45+ 1+ 2
	maxfiles	現在の設定値 + 5
プロセス	nproc	6+ 3+ 4

### (2) Solaris の場合

システムリソース	パラメーター	見積もり
ファイルシステム	rlim_fd_cur	現在の設定値 + 5
プロセス	max_nprocs	6+ 3+ 4

### (3) AIX の場合

AIX の場合は、カーネルパラメーターの調整は不要です。

### (4) Linux の場合

Linux の場合は、カーネルパラメーターの調整は不要です。

### (5) Linux (IPF) の場合

Linux (IPF) の場合は、カーネルパラメーターの調整は不要です。

## 付録 D SNMP エージェントの前提とするパッチ，プロセス（サービス）一覧

SNMP エージェントが前提とするパッチとプロセス（サービス）を次に示します。これらのパッチとプロセス（サービス）がインストールされていない場合、不正な MIB 値を応答したり、MIB 値が取得できなかったりするなどの問題が発生します。そのため、パッチとプロセス（サービス）をインストールしてください。記載のない OS については、前提とするパッチとプロセス（サービス）はありません。

### 注

ここで前提とするパッチは、必ずしも前提となるプロセス（サービス）を修正するパッチではありません。

OS	前提とするパッチ	前提とするプロセス（サービス）
HP-UX 11i V2 (IPF)	PHSS_34609, または後継パッチファイル	SNMP エージェントは HP-UX (IPF) が提供する mib2agt, hp-unixagt プロセスと連携しています。
HP-UX 11i V3 (IPF)	前提とするパッチはありません。	
AIX 5L V5.3	bos.net.tcp.client5.3.0.60 (または後継パッチファイル)	snmpd, hostmibd, snmpmibd, aixmibd, dpid2
AIX V6.1	前提とするパッチはありません。	
Red Hat Enterprise Linux AS4/ES4 (x86, AMD64&EM64T)	net-snmp-5.1.2-13.el4_7.1 net-snmp-libs-5.1.2-13.el4_7.1 net-snmp-utils-5.1.2-13.el4_7.1	snmpd (RPM パッケージ名は net-snmp です)
Red Hat Enterprise Linux AS4 (IPF)	上記ファイルまたは後継パッチファイル	
Red Hat Enterprise Linux 5.1 (x86, AMD64&EM64T)	net-snmp-5.3.1-24.el5 net-snmp-libs-5.3.1-24.el5	snmpd (RPM パッケージ名は net-snmp です)
Red Hat Enterprise Linux 5.2 (x86, AMD64&EM64T)	net-snmp-utils-5.3.1-24.el5 上記ファイルまたは後継パッチファイル	
Solaris 9	117168-01, または後継パッチファイル	snmpdx
Solaris 10	118373-01, または後継パッチファイル	snmpd, snmpdx

---

## 付録 E IBM RISC システム /6000 SP2(RS/6000 SP2) へのインストールとアンインストール

SNMP エージェントを IBM RISC システム /6000 SP2(RS/6000 SP2) にインストールする方法およびアンインストールする方法について説明します。

### 付録 E.1 IBM RISC システム /6000 SP2(RS/6000 SP2) へのインストール

ここでは、IBM RISC システム /6000 SP2(RS/6000 SP2) へ SNMP エージェントをインストールする方法を説明します。

SNMP エージェントのインストールに必要な作業を次に示します。

#### コントロールワークステーションへのインストール

コントロールワークステーションへのインストールは Hitachi PP Installer を使用します。

#### 各ノードへのコピー配布

各ノードへのコピー配布の手順については、「(1) 各ノードへのコピー配布」を参照してください。

#### 各ノードでのインストールの実行

各ノードでのインストールの手順については、「(2) 各ノードでのインストールの実行」を参照してください。

#### (1) 各ノードへのコピー配布

コントロールワークステーションにインストールしたファイルを各ノードに配布する手順を次に示します。

##### 1. ファイルをアーカイブする。

コントロールワークステーション内にインストールした次のファイルをアーカイブします (アーカイブファイル 1.ar)。ファイルのアーカイブには tar コマンドを使用します。

- /usr/OV/newconfig/EAGENT-RUN 配下のすべてのファイル
- /usr/OV/prg\_samples/eagent 配下のすべてのファイル
- /usr/OV/snmp\_mibs/eagent 配下のすべてのファイル
- /usr/CM2/ESA 配下のすべてのファイル

##### (例)

アーカイブファイル名 /tmp/esa1 を作成します。

```
# tar cvf /tmp/esa1.ar /usr/OV/newconfig/EAGENT-RUN
/usr/OV/prg_samples/eagent /usr/OV/snmp_mibs/eagent /usr/CM2/ESA
```

同様に、tar コマンドを使用して、次のファイルをアーカイブします（アーカイブファイル 2）。

- /etc/.hitachi/before/before\_112B
- /etc/.hitachi/after/after\_112B
- /etc/.hitachi/remove/remove\_112B

（例）

アーカイブファイル名 /tmp/esa2 を作成します。

```
# tar cvf /tmp/esa2.ar /etc/.hitachi/before/before_112B
/etc/.hitachi/after/after_112B /etc/.hitachi/remove/remove_112B
```

## 2. アーカイブファイルを各ノードへのコピー配布する。

手順 1 で作成したアーカイブファイルを各ノードへコピー配布します。コピー配布には rcp コマンドを使用します。

（例）

ノード名 host1 にアーカイブファイルを配布します。

```
# rcp /tmp/esa1.ar host1:/tmp/esa1.ar
# rcp /tmp/esa2.ar host1:/tmp/esa2.ar
```

## 3. アーカイブファイルを削除する。

手順 1 で作成したコントロールワークステーションのアーカイブファイルを使用して削除します。アーカイブファイルの削除には rm コマンドを使用します。

（例）

```
# rm /tmp/esa1.ar
# rm /tmp/esa2.ar
```

## （2）各ノードでのインストールの実行

各ノードで SNMP エージェントをインストールする手順を次に示します。

### 1. ログインする。

各ノードにスーパーユーザーでログインします。スーパーユーザー以外でログインした場合は、su コマンドを使用してユーザーをスーパーユーザーに変更します。

### 2. アーカイブファイル 2 を回復する。

各ノードにコピー配布したアーカイブファイル 2 を回復します。

（例）

```
# tar xvf /tmp/esa2.ar
```

### 3. インストールシェルを実行する。

各ノードで /etc/.hitachi/before/before\_112B をスーパーユーザーで実行します。

（例）

```
# /etc/.hitachi/before/before_112B
```

インストールの結果は /tmp/esa.log に出力されます。

4. アーカイブファイル 1 を回復する。

各ノードにコピー配布したアーカイブファイル 1 を回復します。

(例)

```
# tar xvf /tmp/esa1.ar
```

5. ディレクトリを作成する。

各ノードで次のディレクトリを作成します。

- /usr/OV/bin
- /usr/OV/conf
- /etc/srconf/agt
- /etc/SnmpAgent.d

6. インストールシェルを実行する。

各ノードで /etc/.hitachi/after/after\_112B をスーパーユーザーで実行します。

(例)

```
# /etc/.hitachi/after/after_112B
```

インストールの結果は /tmp/esa.log に出力されます。

7. 動作を確認する。

各ノードで SNMP エージェントが起動されていることを確認します。

(例)

```
# /usr/CM2/ESA/bin/snmpcheck
```

/usr/CM2/ESA/bin/snmpcheck コマンドの実行結果に snmpdm , hp\_unixagt , trapdestagt , および htc\_unixagt1 が running と表示されていることを確認します。

8. アーカイブファイルを削除する。

各ノードにコピー配布したアーカイブファイルを削除します。

(例)

```
# rm /tmp/esa1.ar  
# rm /tmp/esa2.ar
```

## 付録 E.2 IBM RISC システム /6000 SP2(RS/6000 SP2) からのアンインストール

ここでは、IBM RISC システム /6000 SP2(RS/6000 SP2) へインストールした SNMP エージェントを削除する方法について説明します。

次の手順で各ノードのプログラムを削除します。

1. アンインストールシェルを実行する。

各ノードで、`/etc/.hitachi/remove/remove_112B` をスーパーユーザーで実行します。

(例)

```
# /etc/.hitachi/remove/remove_112B
```

アンインストールの結果は `/tmp/esa.log` に出力されます。

2. インストールシェルおよびアンインストールシェルのファイルを削除する。

各ノードで次のファイルを削除します。

- `/etc/.hitachi/after/after_112B`
- `/etc/.hitachi/before/before_112B`
- `/etc/.hitachi/remove/remove_112B`

(例)

```
# rm /etc/.hitachi/after/after_112B
# rm /etc/.hitachi/before/before_112B
# rm /etc/.hitachi/remove/remove_112B
```

---

## 付録 F 各バージョンの変更内容

### (1) 09-00 の変更内容

- HP-UX のサポートを中止した。
- mib2agt プロセスの提供を中止した。
- SMF(Service Management Facility) が適用された Solaris 10 版をサポートした。
- SNMP エージェントが稼働するシステムを追加した。
- AIX 版では新規に起動シェルスクリプト (/usr/CM2/ESA/bin/esa) を導入した。
- ログ、16 進パケットダンプおよび VarBind トレースを常時出力に変更した。これに伴い、SR\_LOG\_DIR、SNMP\_HTC\_SNMPD\_LOG\_SIZE 環境変数および SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数を追加した。
- Solaris 版のスワップ空間サイズ取得方法の変更に伴い、SNMP\_HTC\_SOLARIS\_SWAP\_RESERVED 環境変数を追加した。
- AIX 版の SMT 環境下での CPU 利用率取得方法の変更に伴い、SNMP\_HTC\_AIX\_CPU\_SMT 環境変数を追加した。
- AIX 版の物理メモリー取得方法の変更に伴い、SNMP\_HTC\_AIX\_EXCEPT\_FILECACHE 環境変数を追加した。
- coldStart トラップの送信タイミングの変更に伴い、SNMP\_HTC\_INIT\_WAIT\_TIME 環境変数を追加した。

### (2) 08-00 の変更内容

- fileSystem グループについても /etc/SnmpAgent.d/esafsys.conf の設定が有効になることを追加した (AIX, Linux, Linux(IPF) のみ)。
- Linux, Linux (IPF) の SNMP 受信ポート番号を 22161/udp にしました。

### (3) 07-50 の変更内容

- /etc/srconf/agt/naa.cnf ファイルを使用してネイティブエージェントアダプターが、ネイティブエージェントに対して送信する GET リクエスト /SET リクエストで、使用するコミュニティ名を設定する方法を追加した。
- 日立企業固有 MIB オブジェクトに次のグループを追加した。
  - fileSystem64 グループ
  - diskBusyAvail グループ
- 日立企業固有 MIB オブジェクトの cpuUtil グループに次のオブジェクトを追加した。
  - cpuUtilTotalUser
  - cpuUtilTotalSystem
  - cpuUtilTotalWio
  - cpuUtilTotalIdle
- snmpcmdchk コマンドを追加した。
- Linux (IPF) の記述を追加した。

#### (4) 07-10 の変更内容

- 複数の拡張 MIB 定義ファイルを設定できる機能を追加した。
- 日立固有 MIB オブジェクトに次のグループを追加した。  
systemInfo64 グループ, virtualMemory64Ex グループ, process64 グループ,  
processor64 グループ, diskInfo64 グループ, swapSystem64 グループ, および  
diskBusyInfo グループ
- 障害発生時に必要な資料を採取する `jp1esalog.sh.def` コマンドを追加した。
- `htc_monagt1` コマンドに次の引数を追加した。  
-s : CPU 利用時間情報のインターバル時間を設定できる  
-d : ディスクビジー時間取得のインターバル時間を設定できる
- `snmpstart` コマンドの引数に `-e` を追加し, 読み込まれていない拡張 MIB ファイルを  
読み込めるようにした。
- SNMP エージェントのファイル (HP-UX (IPF) の場合) を追加した。

#### (5) 07-00 の変更内容

- `cpuUtil` グループをサポートした。
- `virtualMemory64` グループをサポートした。
- ネイティブエージェントアダプターに送受信パケット取得機能を追加した。
- コミュニティ名不正時にその SNMP リクエストの送信元 IP アドレスとコミュニティ  
名をログに取得する機能を追加した。

#### (6) 06-71 の変更内容

- `DiskBusy` グループ, および `SwapSystem` グループをサポートした。

#### (7) 06-50 の変更内容

- Oracle インテリジェントエージェント連携機能を追加した。
- パケットダンプ取得機能を追加した。
- Turbo Linux, Compaq Tru64 UNIX のサポートにより, MIB オブジェクト一覧に項目  
を追加した。
- `extsubagt` コマンドのオプションを追加した。

## 付録 G (3020-3-T04-01) の変更内容

表 G-1 (3020-3-T04-01) の変更内容

#	変更箇所	追加・変更内容																																						
1	1.2.2(1)	<p>表 1-2 SNMP トラップの種類別と標準トラップ番号一覧 [訂正前]</p> <table border="1"> <thead> <tr> <th>SNMP トラップの種類</th> <th>標準トラップ番号</th> <th>意味</th> </tr> </thead> <tbody> <tr> <td rowspan="5">一般トラップ</td> <td>0</td> <td>ColdStart</td> </tr> <tr> <td>1</td> <td>WarmStart</td> </tr> <tr> <td>2</td> <td>LinkDown</td> </tr> <tr> <td>3</td> <td>LinkUp</td> </tr> <tr> <td>4</td> <td>AuthenticationFailure</td> </tr> <tr> <td>5</td> <td>EgpNeighborLoss</td> </tr> <tr> <td>拡張トラップ</td> <td>6</td> <td>EnterpriseSpecific</td> </tr> </tbody> </table> <p>[訂正後]</p> <table border="1"> <thead> <tr> <th>SNMP トラップの種類</th> <th>標準トラップ番号</th> <th>意味</th> </tr> </thead> <tbody> <tr> <td rowspan="5">一般トラップ※</td> <td>0</td> <td>ColdStart</td> </tr> <tr> <td>1</td> <td>WarmStart</td> </tr> <tr> <td>2</td> <td>LinkDown</td> </tr> <tr> <td>3</td> <td>LinkUp</td> </tr> <tr> <td>4</td> <td>AuthenticationFailure</td> </tr> <tr> <td>5</td> <td>EgpNeighborLoss</td> </tr> <tr> <td>拡張トラップ</td> <td>6</td> <td>EnterpriseSpecific</td> </tr> </tbody> </table> <p>注 SNMP エージェントが発行する一般トラップは、ColdStart トラップおよび、AuthenticationFailure トラップです。その他の一般トラップは発行しません。</p>	SNMP トラップの種類	標準トラップ番号	意味	一般トラップ	0	ColdStart	1	WarmStart	2	LinkDown	3	LinkUp	4	AuthenticationFailure	5	EgpNeighborLoss	拡張トラップ	6	EnterpriseSpecific	SNMP トラップの種類	標準トラップ番号	意味	一般トラップ※	0	ColdStart	1	WarmStart	2	LinkDown	3	LinkUp	4	AuthenticationFailure	5	EgpNeighborLoss	拡張トラップ	6	EnterpriseSpecific
SNMP トラップの種類	標準トラップ番号	意味																																						
一般トラップ	0	ColdStart																																						
	1	WarmStart																																						
	2	LinkDown																																						
	3	LinkUp																																						
	4	AuthenticationFailure																																						
5	EgpNeighborLoss																																							
拡張トラップ	6	EnterpriseSpecific																																						
SNMP トラップの種類	標準トラップ番号	意味																																						
一般トラップ※	0	ColdStart																																						
	1	WarmStart																																						
	2	LinkDown																																						
	3	LinkUp																																						
	4	AuthenticationFailure																																						
5	EgpNeighborLoss																																							
拡張トラップ	6	EnterpriseSpecific																																						
2	2.2.1(2)(a)	<p>表 2-1 SNMP エージェントが MIB 値の取得で使用するコマンド [訂正前]</p> <table border="1"> <thead> <tr> <th>適用OS</th> <th>MIB 値の取得で使用するコマンド</th> <th>コマンドの使用方法</th> </tr> </thead> <tbody> <tr> <td rowspan="6">Solaris</td> <td>/etc/prtconf</td> <td>prtconf</td> </tr> <tr> <td rowspan="4">/usr/bin/sar</td> <td>sar 5 1</td> </tr> <tr> <td>sar -r 5</td> </tr> <tr> <td>sar -d 5</td> </tr> <tr> <td>sar -d 300 1</td> </tr> <tr> <td>/etc/swap -s</td> <td>swap -s</td> </tr> <tr> <td>/usr/bin/pagesize</td> <td>pagesize</td> </tr> <tr> <td>/usr/bin/mpstat</td> <td>mpstat 300 2</td> </tr> </tbody> </table>	適用OS	MIB 値の取得で使用するコマンド	コマンドの使用方法	Solaris	/etc/prtconf	prtconf	/usr/bin/sar	sar 5 1	sar -r 5	sar -d 5	sar -d 300 1	/etc/swap -s	swap -s	/usr/bin/pagesize	pagesize	/usr/bin/mpstat	mpstat 300 2																					
適用OS	MIB 値の取得で使用するコマンド	コマンドの使用方法																																						
Solaris	/etc/prtconf	prtconf																																						
	/usr/bin/sar	sar 5 1																																						
		sar -r 5																																						
		sar -d 5																																						
		sar -d 300 1																																						
	/etc/swap -s	swap -s																																						
/usr/bin/pagesize	pagesize																																							
/usr/bin/mpstat	mpstat 300 2																																							

#	変更箇所	追加・変更内容																									
		<p>[ 訂正後 ]</p> <table border="1"> <thead> <tr> <th>適用OS</th> <th>MIB値の取得で使用するコマンド</th> <th>コマンドの使用方法</th> </tr> </thead> <tbody> <tr> <td rowspan="6">Solaris</td> <td>/etc/prtconf</td> <td>prtconf</td> </tr> <tr> <td rowspan="4">/usr/bin/sar</td> <td>sar 5 1</td> </tr> <tr> <td>sar -r 5</td> </tr> <tr> <td>sar -d 5</td> </tr> <tr> <td>sar -d 300 1</td> </tr> <tr> <td>/etc/swap</td> <td>swap -s</td> </tr> <tr> <td>/usr/bin/pagesize</td> <td>pagesize</td> </tr> <tr> <td>/usr/bin/mpstat</td> <td>mpstat 300 2</td> </tr> </tbody> </table>	適用OS	MIB値の取得で使用するコマンド	コマンドの使用方法	Solaris	/etc/prtconf	prtconf	/usr/bin/sar	sar 5 1	sar -r 5	sar -d 5	sar -d 300 1	/etc/swap	swap -s	/usr/bin/pagesize	pagesize	/usr/bin/mpstat	mpstat 300 2								
適用OS	MIB値の取得で使用するコマンド	コマンドの使用方法																									
Solaris	/etc/prtconf	prtconf																									
	/usr/bin/sar	sar 5 1																									
		sar -r 5																									
		sar -d 5																									
		sar -d 300 1																									
	/etc/swap	swap -s																									
/usr/bin/pagesize	pagesize																										
/usr/bin/mpstat	mpstat 300 2																										
3	2.2.1(2)	<p>(a) MIB 値の取得で使用するコマンド</p> <p>[ 訂正前 ]</p> <p>注 2 Linux および Linux (IPF) で SNMP エージェントを使用するときは, mpstat コマンドは syststat パッケージに入っています。</p> <p>[ 訂正後 ]</p> <p>注 2 Linux および Linux (IPF) で SNMP エージェントを使用するときは, mpstat コマンドは sysstat パッケージに入っています。</p>																									
4	2.2.2	<p>(2) アンインストール時の注意事項</p> <p>[ 追加 ]</p> <ul style="list-style-type: none"> <li>• /etc/SnmpAgent.d/esalocale.conf</li> </ul>																									
5	2.2.3(1)	<p>表 2-2 パージョンアップインストールで上書きされないファイル一覧</p> <p>[ 訂正前 ]</p> <table border="1"> <thead> <tr> <th rowspan="2">種類</th> <th rowspan="2">パス名</th> <th rowspan="2">ファイル名</th> <th colspan="4">適用OS</th> </tr> <tr> <th>Solaris</th> <th>AIX</th> <th>Linux, Linux (IPF)</th> <th>HP-UX (IPF)</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> <td>:</td> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td>ファイルシステム定義ファイル</td> <td>/etc/SnmpAgent.d</td> <td>esafilesys.conf</td> <td>—</td> <td>○</td> <td>○</td> <td>—</td> </tr> </tbody> </table>	種類	パス名	ファイル名	適用OS				Solaris	AIX	Linux, Linux (IPF)	HP-UX (IPF)	:	:	:	:	:	:	:	ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf	—	○	○	—
種類	パス名	ファイル名				適用OS																					
			Solaris	AIX	Linux, Linux (IPF)	HP-UX (IPF)																					
:	:	:	:	:	:	:																					
ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf	—	○	○	—																					

#	変更箇所	追加・変更内容						
		[ 訂正後 ]						
		種類	パス名	ファイル名	適用OS			
					Solaris	AIX	Linux, Linux (IPF)	HP-UX (IPF)
		:	:	:	:	:	:	:
		ファイルシステム定義ファイル	/etc/SnmpAgent.d	esafilesys.conf	○	○	○	○
		動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf	○	○	○	○
6	2.3.1	Solaris 10 の場合 SUNWsmagt, SUNWsasnm, SUNWmibii [ 追加 ] なお, Solaris 10 の最新アップデート版では以下のパッケージに変更されますので, ご注意ください。 SUNWsmmgr, SUNWsmagt, SUNWsmcmd						
7	2.3.1	Solaris 10 の場合 /usr/bin/pkginfo SUNWsmagt SUNWsasnm SUNWmibii [ 追加 ] なお, Solaris 10 の最新アップデート版では次のコマンドで確認してください。 /usr/bin/pkginfo SUNWsmmgr SUNWsmagt SUNWsmcmd						
8	2.6.2	[ 訂正前 ] なお, インターネット標準 MIB-II, HP 企業固有 MIB および日立企業固有 MIB の MIB モジュールがオンラインでアクセスできますので参照してください。それぞれのファイルを次に示します。 [ 訂正後 ] なお, インターネット標準 MIB-II, HP 企業固有 MIB および日立企業固有 MIB の MIB モジュールに関しては, 以下のファイルを参照して下さい。						

#	変更箇所	追加・変更内容
9	2.8	<p>[訂正前]</p> <ul style="list-style-type: none"> <li>共有ディスクの監視に必要な設定 (Linux, Linux(IPF) の場合) SNMP エージェントをクラスタシステムにインストールした場合に、監視マネージャーから共有ディスクのファイルシステム情報が取得できないときがあります。原因は、対象となるファイルシステムが <code>/etc/fstab</code> に記載されていないためです。これは、SNMP エージェントはファイルシステム情報の取得に OS のシステムコールを使用していますが、このシステムコールは <code>/etc/fstab</code> に記載されているファイルシステムだけを対象としているためです。この対策としては、<code>/etc/fstab</code> に共有ディスクのファイルシステム情報を追加します。ここで、<code>/etc/fstab</code> に記載されたファイルシステムは、OS 起動時などに自動マウントされるため、クラスタシステムの場合、共有ディスクが両系にマウントされる現象が発生するおそれがあるので注意が必要です。なお、<code>/etc/fstab</code> の詳細については OS のドキュメントを参照してください。</li> </ul> <p>次に、<code>/etc/fstab</code> の記述例を示します</p> <p>(例)</p> <p>記述例は、共有ディスクを <code>/mnt/test</code> としています。</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre data-bbox="499 710 1199 784">/dev/sdb1    /mnt/test  ignore     defaults   0 0</pre> </div> <p><code>/etc/fstab</code> の第 3 フィールドに、ファイルシステムタイプを <code>ignore</code> とすると、該当ファイルシステムは自動マウントされません。そのため、クラスタ共有ディスクを <code>/etc/fstab</code> へ登録し、その第 3 フィールドを <code>ignore</code> とします。ご使用のシステムで、記述例が有効かどうかは OS のドキュメントなどを参照して確認してください。</p>

#	変更箇所	追加・変更内容
		<p>[訂正後]</p> <ul style="list-style-type: none"> <li>共有ディスクの監視に必要な設定 (Linux, Linux(IPF) の場合)</li> </ul> <p>SNMP エージェントをクラスタシステムにインストールした場合に、監視マネージャーから共有ディスクのファイルシステム情報が取得できないときがあります。原因は、対象となるファイルシステムが /etc/fstab に記載されていないためです。これは、SNMP エージェントはファイルシステム情報の取得に OS のシステムコールを使用していますが、このシステムコールは /etc/fstab に記載されているファイルシステムだけを対象としているためです。この対策としては、/etc/fstab に共有ディスクのファイルシステム情報を追加します。SNMP エージェントは、共有ディスクを監視するにあたり、/etc/fstab に記述された共有ディスクのフィールドのうち、第 1 フィールド (プロックスベシャルデバイス) と第 2 フィールド (マウントポイント) のみを意識しており、その他のフィールドの設定値について、設定値が何であるかは意識していません。</p> <p>そのため、/etc/fstab に記述された共有ディスクの他のフィールド部分に関して具体的な設定方法や設定値は、ご使用のクラスタソフトおよび OS のドキュメントを参照してください。</p> <p>例えば、ご使用のクラスタ管理ソフトが HA モニタの場合で、共有ディスクのマウントポイントが /mnt/test であり、その共有ディスクを制御する際の要件として、OS 起動時などに共有ディスクが自動マウントされないようにする場合の、/etc/fstab の設定例を以下に示します。</p> <p>(設定例)</p> <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre>/dev/sdb1 /mnt/test ext3 defaults,noauto 0 0</pre> </div> <p>上記は、HA モニタを使用している場合の設定例になります。設定を行う際は、HA モニタのマニュアルおよびリリースノートをご確認頂き、設定を行ってください。</p> <p>その他のクラスタソフトを使用している場合は、ご使用のクラスタソフトや OS 環境の要件に合わせて適切な設定を行ってください。</p>

#	変更箇所	追加・変更内容
10	2.8	<p>[追加]</p> <ul style="list-style-type: none"> <li>PowerHA(HACMP)を使用する際の設定 SNMP エージェントを PowerHA(HACMP) 環境で動作させる場合には、SNMP エージェントの SNMP リクエスト受信ポートを 161/udp から空いている UDP ポートに変更して、AIX が提供する snmpd プロセスが 161/udp を使用するよう設定してください。 変更の理由としては、PowerHA(HACMP) 環境下では AIX が提供するプロセスが SNMP Request を使用して情報の交換をしているためです。</li> </ul> <p>以下の例では、SNMP エージェントの SNMP 受信ポートを 8161/udp に変更しています。</p> <p>(1) NNMi または NNM の SNMP リクエストポートの変更 SNMP リクエストポートの変更については、NNMi をご使用の場合は NNMi のヘルプを、また NNM をご使用の場合は NNM のヘルプを参照して下さい。</p> <p>(2) SNMP エージェントの SNMP 受信ポートの変更 以下の作業をスーパーユーザーで実行します。</p> <ol style="list-style-type: none"> <li>SNMP エージェントを停止します。 /usr/CM2/ESA/bin/snmpstop を実行します。</li> <li>AIX の提供する snmpd の SNMP 受信ポートの変更 <ul style="list-style-type: none"> <li>AIX 5L V5.3/V6.1 でネイティブエージェントが SNMP v1 エージェントの場合 /etc/services ファイルの snmp 列の値が 161/udp になっているのを確認してください。161/udp 以外の場合は変更してください。</li> <li>AIX 5L V5.3/V6.1 でネイティブエージェントが SNMP v3 エージェントの場合 /usr/CM2/ESA/opt/SnmpNative を以下のように変更してください。 SNMP_NATIVE_OPTIONS="-p 161"</li> <li>/etc/environment ファイルに「SNMP_PORT=」を定義している場合は以下のように変更してください。 SNMP_PORT=161</li> </ul> </li> <li>SNMP エージェントの SNMP 受信ポートを変更します。 以下の例では SNMP エージェントの SNMP 受信ポートを 8161/udp に変更する場合で説明します。 ポート番号は 8161/udp である必要はありません。 スーパーユーザーで以下の作業を実行します。 /usr/CM2/ESA/opt/SnmpMaster に次の 2 行を追加してください。 SR_SNMP_TEST_PORT=8161 export SR_SNMP_TEST_PORT</li> <li>SNMP エージェントのネイティブエージェントアダプターの SNMP リクエスト送信ポート変更 スーパーユーザーで /usr/CM2/ESA/opt/SnmpNaa ファイルを編集します。 以下の 2 行をファイルの最後に追加してください。 SNMP_NAA_OPTIONS="-port 161" export SNMP_NAA_OPTIONS</li> </ol> <p>上記設定を有効にするため、以下のコマンドを実行して SNMP エージェントを起動して下さい。 /usr/CM2/ESA/bin/snmpstart</p>

#	変更箇所	追加・変更内容																		
11	3.1.1(3)	<p>表 3-2 プロセスの起動オプションを指定する環境変数と指定できるオプション [ 訂正前 ]</p> <table border="1"> <thead> <tr> <th>SNMPエージェントが提供するプロセス</th> <th>起動オプション環境変数</th> <th>指定できるオプション*</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td>htc_monagt1</td> <td>SNMP_MONAGT1_OPTIONS</td> <td>-i, -t, -s, -d</td> </tr> </tbody> </table> <p>[ 訂正後 ]</p> <table border="1"> <thead> <tr> <th>SNMPエージェントが提供するプロセス</th> <th>起動オプション環境変数</th> <th>指定できるオプション*</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td>htc_monagt1</td> <td>SNMP_HTCMONAGT1_OPTIONS</td> <td>-i, -t, -s, -d</td> </tr> </tbody> </table>	SNMPエージェントが提供するプロセス	起動オプション環境変数	指定できるオプション*	:	:	:	htc_monagt1	SNMP_MONAGT1_OPTIONS	-i, -t, -s, -d	SNMPエージェントが提供するプロセス	起動オプション環境変数	指定できるオプション*	:	:	:	htc_monagt1	SNMP_HTCMONAGT1_OPTIONS	-i, -t, -s, -d
SNMPエージェントが提供するプロセス	起動オプション環境変数	指定できるオプション*																		
:	:	:																		
htc_monagt1	SNMP_MONAGT1_OPTIONS	-i, -t, -s, -d																		
SNMPエージェントが提供するプロセス	起動オプション環境変数	指定できるオプション*																		
:	:	:																		
htc_monagt1	SNMP_HTCMONAGT1_OPTIONS	-i, -t, -s, -d																		
12	3.2.2	<p>[ 追加 ]            なお、ネイティブエージェント snmpd が SNMP v1 エージェントの場合は、ネイティブエージェント snmpd が、SNMP 受信ポートとして 161/udp を使用して、SNMP エージェントの SNMP 受信ポートは 161/udp 以外のポートを使用する必要があります。            環境変数 SNMP_NATIVE_OPTIONS は、ネイティブエージェント snmpd が SNMP v3 エージェントの場合に使用します。</p>																		
13	3.4.1	<p>[ 訂正前 ]</p> <ul style="list-style-type: none"> <li>• /etc/SnmpAgent.d/esafilesys.conf ( AIX , Linux , Linux (IPF) の場合 )</li> </ul> <p>[ 訂正後 ]</p> <ul style="list-style-type: none"> <li>• /etc/SnmpAgent.d/esafilesys.conf</li> </ul>																		
14	3.4.1	<p>[ 追加 ]</p> <ul style="list-style-type: none"> <li>• /etc/SnmpAgent.d/esalocale.conf</li> </ul>																		
15	3.5	<p>[ 追加 ]</p> <ul style="list-style-type: none"> <li>• JP1/Cm2/SSO から論理 IP アドレスを監視する際の注意事項               <ul style="list-style-type: none"> <li>• AIX の場合                    クラスタソフトとして日立の HA モニタを使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。</li> <li>• Linux の場合                    クラスタソフトとして日立の HA モニタ、CLUSTERPRO X、Veritas Cluster Server を使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。</li> <li>• Linux (IPF) の場合                    クラスタソフトとして日立の HA モニタを使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。</li> <li>• HP-UX (IPF) の場合                    クラスタソフトとして日立の HA モニタおよび HP Serviceguard を使用した場合、JP1/Cm2/SSO のリソースブラウザのネットワークサマリに論理 IP アドレスの情報は表示されません。</li> </ul> </li> </ul>																		

#	変更箇所	追加・変更内容
16	3.5.2	<p>・使用中の物理メモリー容量の取得についての注意事項</p> <p>[ 訂正前 ] 環境変数定義ファイル SnpHtcMonagt1 ファイルで SNMP_HTC_AIX_EXCEPT_FILECACHE 環境変数に Y を指定すると、ファイルキャッシュ値を含まない使用中の物理メモリー容量を取得できます。</p> <p>[ 訂正後 ] 環境変数定義ファイル SnpHpunix ファイルで SNMP_HTC_AIX_EXCEPT_FILECACHE 環境変数に Y を指定すると、ファイルキャッシュ値を含まない使用中の物理メモリー容量を取得できます。</p>
17	4.2.2	<p>(1) ComputerSystem グループ</p> <p>[ 追加 ]</p> <p>注 6 使用中の物理メモリー容量の取得についての注意事項を次に示します。 AIX では使用中の物理メモリーをファイルキャッシュとして利用することで、ファイルアクセスを向上させています。そのため、SNMP エージェントが取得する使用中の物理メモリー容量には、デフォルトでファイルキャッシュ値が含まれています。</p> <p>環境変数定義ファイル SnpHpunix ファイルで SNMP_HTC_AIX_EXCEPT_FILECACHE 環境変数に Y を指定すると、ファイルキャッシュ値を含まない使用中の物理メモリー容量を取得できます。 環境変数定義ファイルのパスは、「付録 A SNMP エージェントのファイルの一覧」を参照してください。</p> <p>SNMP_HTC_AIX_EXCEPT_FILECACHE 環境変数の指定例を次に示します。</p> <p>(例) SNMP_HTC_AIX_EXCEPT_FILECACHE=Y export SNMP_HTC_AIX_EXCEPT_FILECACHE</p>
18	4.3.2(20)	<p>・注意事項</p> <p>・fileSystem64 グループで監視できるファイルシステムのタイプを次に示します。 なお、NFS を監視する場合、NFS サーバ側でファイルシステムを監視するようにしてください。</p> <p>[ 削除 ] なお、タイプが nfs, swap のファイルシステムは監視できません。</p>

#	変更箇所	追加・変更内容																																																																																																																														
19	4.3.3(11)	<p>表 4-67 日立企業固有 MIB オブジェクトの実装状況 (cpuUtil グループ) [訂正前]</p> <table border="1"> <thead> <tr> <th colspan="2" rowspan="3">オブジェクト名</th> <th colspan="6">MIBオペレーション</th> </tr> <tr> <th colspan="2">Solarisの場合</th> <th colspan="2">AIXの場合</th> <th colspan="2">Linux, Linux (IPF)の場合</th> </tr> <tr> <th>get</th> <th>set</th> <th>get</th> <th>set</th> <th>get</th> <th>set</th> </tr> </thead> <tbody> <tr> <td rowspan="6">cpuUtilTable. cpuUtilEntry</td> <td>cpuUtilCpuNum</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilCpuUser</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilCpuSystem</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilCpuWio</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>cpuUtilCpuIdle</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilTime</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> </tbody> </table> <p>[訂正後]</p> <table border="1"> <thead> <tr> <th colspan="2" rowspan="3">オブジェクト名</th> <th colspan="6">MIBオペレーション</th> </tr> <tr> <th colspan="2">Solarisの場合</th> <th colspan="2">AIXの場合</th> <th colspan="2">Linux, Linux (IPF)の場合</th> </tr> <tr> <th>get</th> <th>set</th> <th>get</th> <th>set</th> <th>get</th> <th>set</th> </tr> </thead> <tbody> <tr> <td rowspan="6">cpuUtilTable. cpuUtilEntry</td> <td>cpuUtilNum</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilUser</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilSystem</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilWio</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>—</td> <td>—</td> </tr> <tr> <td>cpuUtilIdle</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> <tr> <td>cpuUtilTime</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> <td>○</td> <td>—</td> </tr> </tbody> </table>	オブジェクト名		MIBオペレーション						Solarisの場合		AIXの場合		Linux, Linux (IPF)の場合		get	set	get	set	get	set	cpuUtilTable. cpuUtilEntry	cpuUtilCpuNum	○	—	○	—	○	—	cpuUtilCpuUser	○	—	○	—	○	—	cpuUtilCpuSystem	○	—	○	—	○	—	cpuUtilCpuWio	○	—	○	—	—	—	cpuUtilCpuIdle	○	—	○	—	○	—	cpuUtilTime	○	—	○	—	○	—	オブジェクト名		MIBオペレーション						Solarisの場合		AIXの場合		Linux, Linux (IPF)の場合		get	set	get	set	get	set	cpuUtilTable. cpuUtilEntry	cpuUtilNum	○	—	○	—	○	—	cpuUtilUser	○	—	○	—	○	—	cpuUtilSystem	○	—	○	—	○	—	cpuUtilWio	○	—	○	—	—	—	cpuUtilIdle	○	—	○	—	○	—	cpuUtilTime	○	—	○	—	○	—
オブジェクト名		MIBオペレーション																																																																																																																														
		Solarisの場合			AIXの場合		Linux, Linux (IPF)の場合																																																																																																																									
		get	set	get	set	get	set																																																																																																																									
cpuUtilTable. cpuUtilEntry	cpuUtilCpuNum	○	—	○	—	○	—																																																																																																																									
	cpuUtilCpuUser	○	—	○	—	○	—																																																																																																																									
	cpuUtilCpuSystem	○	—	○	—	○	—																																																																																																																									
	cpuUtilCpuWio	○	—	○	—	—	—																																																																																																																									
	cpuUtilCpuIdle	○	—	○	—	○	—																																																																																																																									
	cpuUtilTime	○	—	○	—	○	—																																																																																																																									
オブジェクト名		MIBオペレーション																																																																																																																														
		Solarisの場合		AIXの場合		Linux, Linux (IPF)の場合																																																																																																																										
		get	set	get	set	get	set																																																																																																																									
cpuUtilTable. cpuUtilEntry	cpuUtilNum	○	—	○	—	○	—																																																																																																																									
	cpuUtilUser	○	—	○	—	○	—																																																																																																																									
	cpuUtilSystem	○	—	○	—	○	—																																																																																																																									
	cpuUtilWio	○	—	○	—	—	—																																																																																																																									
	cpuUtilIdle	○	—	○	—	○	—																																																																																																																									
	cpuUtilTime	○	—	○	—	○	—																																																																																																																									
20	5.	<p>systemtrap [追加] systemtrap コマンドは、監視プログラムとして JP1 Version 8 以前の JP1/Cm2/ SNMP System Observer を使用する場合の旧バージョン互換のために提供しています。systemtrap コマンドの詳細については、JP1 Version 8 以前の JP1/Cm2/ SNMP System Observer のマニュアルを参照して下さい。</p>																																																																																																																														

#	変更箇所	追加・変更内容																						
21	付録 A.1	<p>[ 追加 ]</p> <table border="1"> <thead> <tr> <th>種類</th> <th>パス名</th> <th>ファイル名</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td>動作ロケール定義ファイル</td> <td>/etc/SnmpAgent.d</td> <td>esalocale.conf</td> </tr> </tbody> </table>	種類	パス名	ファイル名	:	:	:	動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf													
種類	パス名	ファイル名																						
:	:	:																						
動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf																						
22	付録 A.2	<p>[ 訂正前 ]</p> <table border="1"> <thead> <tr> <th>種類</th> <th>パス名</th> <th>ファイル名</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td rowspan="2">起動ファイル</td> <td>/etc/inittabに登録</td> <td>—</td> </tr> <tr> <td>/opt/CM2/ESA/bin</td> <td>esa</td> </tr> </tbody> </table> <p>[ 訂正後 ]</p> <table border="1"> <thead> <tr> <th>種類</th> <th>パス名</th> <th>ファイル名</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td rowspan="2">起動ファイル</td> <td>/etc/inittabに登録</td> <td>—</td> </tr> <tr> <td>/usr/CM2/ESA/bin</td> <td>esa</td> </tr> </tbody> </table>	種類	パス名	ファイル名	:	:	:	起動ファイル	/etc/inittabに登録	—	/opt/CM2/ESA/bin	esa	種類	パス名	ファイル名	:	:	:	起動ファイル	/etc/inittabに登録	—	/usr/CM2/ESA/bin	esa
種類	パス名	ファイル名																						
:	:	:																						
起動ファイル	/etc/inittabに登録	—																						
	/opt/CM2/ESA/bin	esa																						
種類	パス名	ファイル名																						
:	:	:																						
起動ファイル	/etc/inittabに登録	—																						
	/usr/CM2/ESA/bin	esa																						
23	付録 A.3	<p>[ 追加 ]</p> <table border="1"> <thead> <tr> <th>種類</th> <th>パス名</th> <th>ファイル名</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td>動作ロケール定義ファイル</td> <td>/etc/SnmpAgent.d</td> <td>esalocale.conf</td> </tr> </tbody> </table>	種類	パス名	ファイル名	:	:	:	動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf													
種類	パス名	ファイル名																						
:	:	:																						
動作ロケール定義ファイル	/etc/SnmpAgent.d	esalocale.conf																						
24	付録 A.5	<p>[ 訂正前 ]</p> <table border="1"> <thead> <tr> <th>種類</th> <th>パス名</th> <th>ファイル名</th> </tr> </thead> <tbody> <tr> <td rowspan="3">MIB値取得コマンド</td> <td rowspan="3">/opt/CM2/ESA/bin</td> <td>:</td> </tr> <tr> <td>linuxSystem.exe</td> </tr> <tr> <td>:</td> </tr> </tbody> </table> <p>[ 訂正後 ]</p> <table border="1"> <thead> <tr> <th>種類</th> <th>パス名</th> <th>ファイル名</th> </tr> </thead> <tbody> <tr> <td rowspan="3">MIB値取得コマンド</td> <td rowspan="3">/opt/CM2/ESA/bin</td> <td>:</td> </tr> <tr> <td>linuxSwap.exe</td> </tr> <tr> <td>:</td> </tr> </tbody> </table>	種類	パス名	ファイル名	MIB値取得コマンド	/opt/CM2/ESA/bin	:	linuxSystem.exe	:	種類	パス名	ファイル名	MIB値取得コマンド	/opt/CM2/ESA/bin	:	linuxSwap.exe	:						
種類	パス名	ファイル名																						
MIB値取得コマンド	/opt/CM2/ESA/bin	:																						
		linuxSystem.exe																						
		:																						
種類	パス名	ファイル名																						
MIB値取得コマンド	/opt/CM2/ESA/bin	:																						
		linuxSwap.exe																						
		:																						
25	付録 D	<p>[ 訂正前 ]</p> <table border="1"> <thead> <tr> <th>OS</th> <th>前提とするパッチ</th> <th>前提とするプロセス (サービス)</th> </tr> </thead> <tbody> <tr> <td>:</td> <td>:</td> <td>:</td> </tr> <tr> <td>AIX 5L V5.3</td> <td rowspan="2">前提とするパッチはありません</td> <td rowspan="2">snmpd, hostnibd, snmpm ibd, aixnibd, dpid2</td> </tr> <tr> <td>AIX V6.1</td> </tr> </tbody> </table>	OS	前提とするパッチ	前提とするプロセス (サービス)	:	:	:	AIX 5L V5.3	前提とするパッチはありません	snmpd, hostnibd, snmpm ibd, aixnibd, dpid2	AIX V6.1												
OS	前提とするパッチ	前提とするプロセス (サービス)																						
:	:	:																						
AIX 5L V5.3	前提とするパッチはありません	snmpd, hostnibd, snmpm ibd, aixnibd, dpid2																						
AIX V6.1																								

#	変更箇所	追加・変更内容		
		[ 訂正後 ]		
		OS	前提とするパッチ	前提とするプロセス (サービス)
		:	:	:
		AIX 5L V5.3	bos.net.tcp.client5.3.0.60 (または後継パッチファイル)	snmpd, hostmibd, snmpm ibd, aixmibd, dpid2
		AIX V6.1	前提とするパッチはありません。	

---

## 付録 H 用語解説

### (英字)

---

#### enterprise ID

企業を識別するコード(企業名)です。

#### Extensible SNMP Agent

UNIX で動作する日立のエージェントです。

#### MIB オブジェクト

ある特定のタイプまたはクラスの管理情報です。

#### MIB オペレーション

MIB オブジェクトに対する操作です。MIB オペレーションには、Get オペレーション、GetNext オペレーション、および Set オペレーションがあります。

#### MIB 値

MIB オブジェクトの値です。

#### MIB モジュール

複数の MIB オブジェクトをツリー構造にまとめたものです。

#### netmon (ネットワークモニタプロセス)

「ネットワークモニタプロセス (netmon)」を参照してください。

#### NNM または NNMi

ネットワークの構成管理、性能管理および障害管理のためのプログラムです。

#### Oracle インテリジェントエージェント

Oracle が提供している SNMP エージェントです。

#### ovspmd (プロセスマネージャー)

「プロセスマネージャー (ovspmd)」を参照してください。

#### pmd (ポストマスタ)

「ポストマスタ (pmd)」を参照してください。

#### SNMP

Simple Network Management Protocol の略で、インターネット環境でのネットワーク管理プロトコルです。

#### SubManager

ネットワークの階層管理のためのプログラムです。

## (ア行)

---

### 一般トラップ番号

RFC1157 に定義されている SNMP のトラップ番号です。

### エージェント

ネットワーク管理で、管理対象ノードで動作し、ネットワーク資源を管理対象オブジェクトとして管理します。

### エージェントシステム

エージェントが動作するシステムです。

## (カ行)

---

### 階層エージェント

マネージャーシステムで動作するネットワークの階層管理のためのプログラムです。この上位にマネージャーを配置することで多くの階層の管理をします。ただし、このマニュアルでは、特に限定していなければ、ネットワーク階層管理装置の総称として使用します。

### 拡張 MIB オブジェクト

SNMP エージェントが提供する拡張 MIB オブジェクト定義機能を使用して定義する、ユーザー独自の MIB オブジェクトです。

### 拡張トラップ

SNMP エージェントが提供している `snmptrap` コマンドを使用して発行する、ユーザー独自のトラップです。

### 起動設定情報

Network Node Manager および SubManager の各プロセスに関する設定情報です。起動コマンド (`ovstart`) 実行時、プロセスマネージャーで参照します。設定するファイルは、`/usr/OV/conf/ovsuf` です。

### コミュニティ名

SNMP プロトコルで、エージェントの MIB 値にアクセスするためのパスワードです。

### 固有トラップ番号

ユーザー独自のトラップに割り当てる番号です。企業 ID ごとにユニークな番号にする必要があります。

## (サ行)

---

### サブマネージャー

ネットワーク管理での、マネージャーとエージェントの中間に位置するネットワークの階層管理のためのプログラムです。マネージャーに対してはエージェントとして動作し、エージェントに対してはマネージャーとして動作します。

## サブマネージャーシステム

サブマネージャーが動作するシステムです。

## (ナ行)

---

### 認証失敗 ( 認証エラー )

SNMP で規定されるエージェントへの MIB 操作時のエラーの一つです。コミュニケーション不正時に発生します。

### ネットワークモニタプロセス ( netmon )

Network Node Manager および SubManager のバックグラウンドプロセスの一つです。SNMP 要求、および ICMP エコー要求を使用してポーリングをし、ネットワーク上のノードを検出します。

## (ハ行)

---

### 標準 MIB オブジェクト

RFC1213 で定義されているインターネット標準の MIB オブジェクトです。

### プロセスマネージャー ( ovspmd )

Network Node Manager および SubManager のバックグラウンドプロセスの一つです。配下のプロセスの開始、終了、およびプロセス状態を監視します。

### ポストマスタ ( pmd )

Network Node Manager および SubManager のバックグラウンドプロセスの一つです。SNMP トラップや SNMP 要求の振り分け機能を持っています。

## (マ行)

---

### マネージャー

ネットワーク管理でのネットワーク管理装置で、管理対象ノードを管理します。

### マネージャーシステム

マネージャーが動作するシステムです。

## (ラ行)

---

### ローカル登録ファイル

Network Node Manager および SubManager のバックグラウンドプロセスの処理についての情報が設定されたファイルです。起動設定情報は、ローカル登録ファイルから作成します。



---

# 索引

## A

---

AddressTranslation グループ 133

## C

---

Cluster グループ 149  
ComputerSystem グループ 142  
cpuUtil グループ 176

## D

---

diskBusyAvail グループ 196  
diskBusyInfo グループ 192  
DiskBusy グループ 174  
diskInfo64 グループ 190  
DiskInfo グループ 171  
disk グループ (AIX の場合) 197

## E

---

enterprise ID 331  
etc/SnmpAgent.d/snmpd.extend ファイルの  
作成例 84  
Extensible SNMP Agent 331  
extsubagt プロセス 252

## F

---

fileSystem64 グループ 193  
FileSystem グループ 145

## G

---

get コミュニティ名の登録 41  
GET リクエストで使用するコミュニティ名  
の指定 55

## H

---

Hitachi PP Installer の起動 25  
Hitachi PP Installer の使用方法 25  
hp\_unixagt プロセス 255  
HP 企業固有 MIB オブジェクト

Cluster グループ 149  
ComputerSystem グループ 142  
FileSystem グループ 145  
ICMP グループ 150  
Ieee8023Mac グループ 149  
Processes グループ 147  
SnmpdConf グループ 151  
Trap グループ 151

HP 企業固有 MIB オブジェクト一覧 141  
HP 企業固有 MIB オブジェクトの構成 141  
HP 企業固有 MIB オブジェクトの実装状況  
152

Cluster グループ 156  
ComputerSystem グループ 152  
FileSystem グループ 153  
ICMP グループ 158  
Ieee8023Mac グループ 156  
Processes グループ 154  
SnmpdConf グループ 159  
Trap グループ 158

HP 企業固有 MIB オブジェクトの内容 142  
htc\_monagt1 プロセス 257  
htc\_unixagt1 プロセス 260  
htc\_unixagt2 プロセス 263  
htc\_unixagt3 プロセス 265

## I

---

IBM RISC システム /6000 SP2(RS/6000  
SP2) からのアンインストール 317  
IBM RISC システム /6000 SP2(RS/6000  
SP2) へのインストール 314  
ICMP グループ 136, 150  
Ieee8023Mac グループ 149  
Interfaces グループ 132  
IP: 44  
IP グループ 133

## J

---

jp1esalog.sh.def コマンド 234

## L

---

LinuxSystem グループ (Linux の場合) 199

## M

---

MIB 2

MIB オブジェクト 331

MIB オブジェクト定義例の説明 88

MIB オブジェクトの定義 61, 82

MIB オペレーション 331

MIB 値 331

MIB 値の取得で使用するコマンド 23

MIB 値の取得で使用するコマンドのインストール (HP-UX (IPF) 以外の場合) 22

MIB ツリーの構成 83

MIB モジュール 331

## N

---

naa.cnf 定義ファイル削除時の注意事項 57

naa.cnf 定義ファイルの仕様についての注意事項 (Solaris の場合) 58

naaagt の SNMP パケット送信ポート変更時の注意事項 57

naaagt プロセス 267

netmon 331

NNM 331

NNMi 331

NNM または NNMi からの SNMP 要求時の動作 52

## O

---

OpConf グループ 162

Oracle インテリジェントエージェント 331

ovspmd 331

## P

---

page グループ (AIX の場合) 198

pipe\_in\_name に渡すデータ 69

pipe\_out\_name に渡すデータ 70

pmd 331

processor64 グループ 184

Processes グループ 147

processor64 グループ 187

Processor グループ 169

Process グループ 166

## S

---

set コミュニティ名の登録 42

SET リクエストで使用するコミュニティ名の指定 56

SNMP 331

SNMP\_HTC\_AIX\_CPU\_SMT 環境変数 258

SNMP\_HTC\_AIX\_EXCEPT\_FILECACHE

環境変数 256

SNMP\_HTC\_AUTH\_LOG 環境変数 250

SNMP\_HTC\_INIT\_WAIT\_TIME 環境変数 250

SNMP\_HTC\_SNMPD\_LOG\_CNT 環境変数 250

SNMP\_HTC\_SNMPD\_LOG\_SIZE 環境変数 250

SNMP\_HTC\_SOLARIS\_SWAP\_RESERVE

D 環境変数 256

SNMP\_HTCMONAGT1\_START 環境変数 258

snmpcheck コマンド 238

snmpcmdchk コマンド 239

SnmpdConf グループ 151

snmpdpm プロセス 248

snmpstart コマンド 240

snmpstop コマンド 241

snmptrap コマンド 242

SNMP エージェント 2

SNMP エージェント運用時の動作 15

SNMP エージェントが管理する MIB オブジェクト 6

SNMP エージェントが提供する環境変数定義ファイル 109

SNMP エージェントが提供するサブエージェントのプロセスと適用 OS 14

SNMP エージェント起動時の動作 15

SNMP エージェントで使用するポート番号 310

SNMP エージェントで使用するポート番号 (HP-UX (IPF), Solaris, AIX の場合) 310

- SNMP エージェントで使用するポート番号 (Linux , Linux (IPF) の場合) 310
  - SNMP エージェントの SNMP 受信ポートの変更 114
  - SNMP エージェントの SNMP 受信ポートの変更方法 114
  - SNMP エージェントのインストール 24
  - SNMP エージェントの運用 107
  - SNMP エージェントの概要 1
  - SNMP エージェントの環境設定 19
  - SNMP エージェントの起動 108
  - SNMP エージェントの起動と停止 108
  - SNMP エージェントの機能 5
  - SNMP エージェントのコマンド 17
  - SNMP エージェントのシステム構成 2
  - SNMP エージェントの前提とするパッチ , プロセス (サービス) 一覧 313
  - SNMP エージェントの停止 112
  - SNMP エージェントの動作環境 3
  - SNMP エージェントのファイルの一覧 290
  - SNMP エージェントのファイルの一覧 (AIX の場合) 294
  - SNMP エージェントのファイルの一覧 (HP-UX (IPF) の場合) 302
  - SNMP エージェントのファイルの一覧 (Linux (IPF) の場合) 305
  - SNMP エージェントのファイルの一覧 (Linux の場合) 299
  - SNMP エージェントのファイルの一覧 (Solaris の場合) 290
  - SNMP エージェントのプロセス 11
  - SNMP エージェントのプロセス構成 (HP-UX (IPF) の場合) 13
  - SNMP エージェントのプロセス構成 (Solaris , AIX , Linux および Linux (IPF) の場合) 12
  - SNMP エージェントを構成するプロセス 11
  - SNMP グループ 138
  - SNMP トラップ 5
  - SNMP トラップ送信先ポート番号の設定 (07-50 以前のバージョンから上書きインストールする場合) 29
  - SNMP トラップの発行 6
  - SNMP トラップの発行の概要 7
  - SNMP トラップの標準トラップ番号 7
  - SNMP トラップ発行時のエージェントアドレス 8
  - SNMP 要求時に実行するコマンドの作成 71
  - SNMP 要求時に処理するファイルの作成 75
  - SNMP 要求の処理 5
  - SR\_LOG\_DIR 環境変数 251
  - SR\_SNMP\_TEST\_PORT 環境変数 (extsubagt) 254
  - SR\_SNMP\_TEST\_PORT 環境変数 (hp\_unixagt) 256
  - SR\_SNMP\_TEST\_PORT 環境変数 (htc\_unixagt1) 262
  - SR\_SNMP\_TEST\_PORT 環境変数 (htc\_unixagt2) 264
  - SR\_SNMP\_TEST\_PORT 環境変数 (htc\_unixagt3) 266
  - SR\_SNMP\_TEST\_PORT 環境変数 (naaagt) 268
  - SR\_SNMP\_TEST\_PORT 環境変数 (snmpdm) 250
  - SR\_SNMP\_TEST\_PORT 環境変数 (trapdestagt) 270
  - SR\_TRAP\_TEST\_PORT 環境変数 250
  - SubManager 331
  - SwapInfo グループ 172
  - SwapSpace グループ 173
  - swapSystem64 グループ 190
  - systemInfo64 グループ 180
  - SystemInfo グループ 162
  - systemtrap コマンド 245
  - System グループ 131
  - system グループ (AIX の場合) 197
  - system グループ (Solaris の場合) 198
- ## T
- 
- TCP グループ 137
  - trapdestagt プロセス 269
  - Trap グループ 151
  - Tru64system グループ (Tru64 の場合) 199

## U

UDP グループ 138

## V

VIEW: 45

virtualMemory64Ex グループ 182

virtualMemory64 グループ 177

VirtualMemory グループ 164

## あ

アンインストール 26

アンインストール時の注意事項 27

## い

一般トラップ 7

一般トラップ番号 332

インストール 22

インストールからセットアップまでの流れ  
20

インストールとアンインストール 22

インストール時の注意事項 32

インストール時の注意事項 (AIX の場合) 35

インストール時の注意事項 (HP-UX (IPF) の  
場合) 37

インストール時の注意事項 (Linux および  
Linux (IPF) の場合) 35

インストール時の注意事項 (Solaris の場合)  
32

インストールの事前準備 22

## う

運用上の注意事項 120

運用上の注意事項 (AIX の場合) 123

運用上の注意事項 (Solaris の場合) 123

## え

エージェント 332

エージェントシステム 332

エンタープライズ ID 8

## か

カーネルパラメーター一覧 312

階層エージェント 332

拡張 MIB オブジェクト 332

拡張 MIB オブジェクトの起動オプション定  
義ファイルの設定方法 81

拡張 MIB オブジェクトの定義 9, 59

拡張 MIB オブジェクトの定義の流れ 60

拡張 MIB オブジェクトの定義例 82

拡張 MIB オブジェクトのマネージャーへの  
コピー 77

拡張 MIB オブジェクトの読み込み 80

SNMP エージェントを停止して拡張  
MIB 定義ファイルを追加する場合 80

SNMP エージェントを停止しないで  
拡張 MIB 定義ファイルを追加する場  
合 80

拡張 MIB 定義ファイルの格納 80

拡張 MIB 定義ファイルの起動の確認 80

拡張 MIB 定義ファイルの作成 62

拡張 MIB 定義ファイルの追加方法 79

拡張機能 9

拡張トラップ 7, 332

拡張トラップの定義 10, 99

拡張トラップの定義方法 99

拡張トラップの利用方法 99

各バージョンの変更内容 318

カスタマイズした定義ファイルのバックア  
ップ 27

環境変数

SNMP\_HTC\_AIX\_CPU\_SMT 258

SNMP\_HTC\_AIX\_EXCEPT\_FILEC

ACHE 256

SNMP\_HTC\_AUTH\_LOG 250

SNMP\_HTC\_INIT\_WAIT\_TIME  
250

SNMP\_HTC\_SNMPD\_LOG\_CNT  
250

SNMP\_HTC\_SNMPD\_LOG\_SIZE  
250

SNMP\_HTC\_SOLARIS\_SWAP\_RES  
ERVED 256

SNMP\_HTCMONAGT1\_START 258

SR\_LOG\_DIR 251  
 SR\_SNMP\_TEST\_PORT  
 (extsubagt) 254  
 SR\_SNMP\_TEST\_PORT  
 (hp\_unixagt) 256  
 SR\_SNMP\_TEST\_PORT  
 (htc\_unixagt1) 262  
 SR\_SNMP\_TEST\_PORT  
 (htc\_unixagt2) 264  
 SR\_SNMP\_TEST\_PORT  
 (htc\_unixagt3) 266  
 SR\_SNMP\_TEST\_PORT (naaagt)  
 268  
 SR\_SNMP\_TEST\_PORT(snmppdm)  
 250  
 SR\_SNMP\_TEST\_PORT  
 (trapdestagt) 270  
 SR\_TRAP\_TEST\_PORT 250

環境変数定義ファイルで指定できる起動オプション 109

## き

起動設定情報 332

## く

クラスタ環境で運用する場合の設定 102

## こ

構成定義ファイル (/etc/SnmpAgent.d/  
 snmpd.conf) のカスタマイズ 39  
 構成定義ファイルの形式 47  
 コマンド 232  
 コマンド一覧 232  
 コマンドとプロセス 231  
 コマンドの詳細 233  
 コマンドを実行したいシステムへのログイン  
 72  
 コミュニティ名 40, 332  
 コミュニティ名の指定方法 43  
   get コミュニティ名 43  
   set コミュニティ名 44  
 オプション 44

コミュニティ名の種類 40  
 コミュニティ名の登録方法 41  
 コミュニティ名不正時のログ取得 282  
 固有トラップ番号 332

## さ

最大接続サブエージェント数の変更 117  
 サブエージェントで動作するプロセス  
 13, 246  
 サブエージェントの再構成 76  
 サブエージェントのログマスク値 275  
 サブマネージャー 332  
 サブマネージャーシステム 333

## し

シェルコマンドの動作確認 75  
 シェルコマンドの引数の確認 75  
 識別番号 69  
 システムの起動時に実行されるファイル 111  
 システムの停止時に実行されるファイル 113  
 情報収集デーモン 246  
 情報収集デーモンと適用 OS 15  
 情報収集デーモンプロセス 15  
 資料の採取 278

## す

スクリプトまたはプログラムの記述 72  
 スクリプト例 100  
 すべての SNMP エージェントの設定 77

## せ

設定ファイルのバックアップとリストア 118  
 セットアップ時の注意事項 105  
 セットアップ時の注意事項 (Linux および  
 Linux (IPF) の場合) 106

## た

対処の手順 272

---

 て
 

---

データ型 64  
 テーブル形式の拡張 MIB オブジェクト 66  
 出口コードのチェック 75

---

 と
 

---

トラップ中に設定するエンタープライズ ID  
 8  
 トラップのあて先 46  
 トラップのあて先の設定 46  
     NNMi および任意のマネージャーの  
     場合 46  
     NNM および SubManager の場合 46  
 トラブルシューティング 271  
 トラブルへの対処方法 283

---

 に
 

---

認証失敗 333  
 認証失敗トラップの送信 46

---

 ね
 

---

ネイティブエージェント 6  
 ネイティブエージェント snmpd の SNMP 受  
 信ポートの変更 (AIX の場合) 115  
 ネイティブエージェントアダプター 50  
 ネイティブエージェントアダプター起動時の  
 動作 52  
 ネイティブエージェントアダプターの機能  
 50  
 ネイティブエージェントアダプターの使用上  
 の注意事項 57  
 ネイティブエージェントアダプターの設定  
 (HP-UX (IPF) 以外の場合) 50  
 ネイティブエージェントアダプターの設定方  
 法 53  
 ネイティブエージェントアダプターの送受信  
 パケットダンプの取得 279  
 ネイティブエージェントアダプターの対象と  
 なるネイティブエージェント 52  
 ネイティブエージェントが提供する MIB オ  
 ブジェクト 6

ネイティブエージェントから取得する MIB  
 グループの一覧 50  
 ネイティブエージェントの MIB オブジェク  
 トの登録 53  
 ネイティブエージェントの起動と停止 113  
 ネイティブエージェントのコミュニティ名  
 55  
 ネイティブエージェントのコミュニティ名設  
 定 (Solaris の場合) 43  
 ネイティブエージェントの再起動時の注意事  
 項 57  
 ネイティブエージェントの設定 53  
 ネットワーク環境設定での一般的な注意事項  
 105  
 ネットワークモニタプロセス 333

---

 は
 

---

バージョンアップインストール 27  
 バックアップとリストア 118

---

 ひ
 

---

引数 72  
 日立企業固有 MIB オブジェクト  
   cpuUtil グループ 176  
   diskBusyAvail グループ 196  
   diskBusyInfo グループ 192  
   DiskBusy グループ 174  
   diskInfo64 グループ 190  
   DiskInfo グループ 171  
   disk グループ 197  
   fileSystem64 グループ 193  
   LinuxSystem グループ (Linux の場  
   合) 199  
   OpConf グループ 162  
   page グループ (AIX の場合) 198  
   process64 グループ 184  
   processor64 グループ 187  
   Processor グループ 169  
   Process グループ 166  
   SwapInfo グループ 172  
   SwapSpace グループ 173  
   swapSystem64 グループ 190

SwapSystem グループ 174  
 systemInfo64 グループ 180  
 SystemInfo グループ 162  
 system グループ 197  
 system グループ (Solaris の場合)  
 198  
 Tru64system グループ (Tru64 の場  
 合) 199  
 virtualMemory64Ex グループ 182  
 virtualMemory64 グループ 177  
 VirtualMemory グループ 164  
 日立企業固有 MIB オブジェクト一覧 160  
 日立企業固有 MIB オブジェクトの構成 160  
 日立企業固有 MIB オブジェクトの実装状況  
 199  
 AIX 固有のグループ 227  
 cpuUtil グループ 214  
 diskBusyAvail グループ 226  
 diskBusyInfo グループ 225  
 DiskBusy グループ 212  
 diskInfo64 グループ 223  
 DiskInfo グループ 210  
 fileSystem64 グループ 226  
 Linux , Linux (IPF) 固有のグループ  
 228  
 OpConf グループ 201  
 process64 グループ 220  
 processor64 グループ 221  
 Processor グループ 207  
 Process グループ 205  
 Solaris 固有のグループ 228  
 SwapInfo グループ 211  
 SwapSpace グループ 212  
 swapSystem64 グループ 224  
 systemInfo64 グループ 217  
 SystemInfo グループ 201  
 Tru64 固有のグループ 229  
 virtualMemory64Ex グループ 218  
 virtualMemory64 グループ 215  
 VirtualMemory グループ 203  
 日立企業固有 MIB オブジェクトの内容 162  
 非テーブル形式の拡張 MIB オブジェクト 63  
 標準 MIB オブジェクト 333

AddressTranslation グループ 133  
 ICMP グループ 136  
 Interfaces グループ 132  
 IP グループ 133  
 SNMP グループ 138  
 System グループ 131  
 TCP グループ 137  
 UDP グループ 138  
 標準 MIB オブジェクト一覧 130  
 標準 MIB オブジェクトの構成 130  
 標準 MIB オブジェクトの実装状況 139  
 System グループ 140  
 標準 MIB オブジェクトの内容 131

## ふ

---

ファイアウォールの通過方向 310  
 複数の拡張 MIB 定義ファイルの設定 78  
 フルバックアップ・フルリストア時の注意事  
 項 118  
 プロセス 246  
 プロセス一覧 246  
 プロセスで指定できる環境変数 111  
 プロセスの起動オプションのカスタマイズお  
 よび環境変数の定義 108  
 プロセスの詳細 247  
 プロセスマネージャー 333  
 プロセスを個別に停止する場合の注意事項  
 112

## ほ

---

ポート番号一覧 310  
 ポストマスタ 333

## ま

---

マスターエージェントで動作するプロセス  
 13, 246  
 マスターエージェントの詳細トレースの取得  
 280  
 マスターエージェントの送受信パケットダン  
 プの取得 278  
 マスターエージェントのログマスク 274  
 マネージャー 333

マネージャーのコマンドによるオブジェクト  
の確認 77  
マネージャーシステム 333  
マネージャーへの MIB の統合 77  
マネージャーへのコミュニティ名の格納 42

## も

---

問題の影響範囲 273

## よ

---

読み書き可能 MIB オブジェクトの追加 55  
読み取り専用 MIB オブジェクトの追加 54

## れ

---

連絡先と設置場所の設定 39

## ろ

---

ローカル登録ファイル 333  
ログ出力オプションの設定 (08-00 以前の  
バージョンから上書きインストールする場  
合) 29  
ログ情報の採取方法 287  
ログの取得 274

# ソフトウェアマニュアルのサービス ご案内

## 1. マニュアル情報ホームページ

ソフトウェアマニュアルの情報をインターネットで公開しています。

URL <http://www.hitachi.co.jp/soft/manual/>

ホームページのメニューは次のとおりです。

マニュアル一覧	日立コンピュータ製品マニュアルを製品カテゴリ、マニュアル名称、資料番号のいずれかから検索できます。
CD-ROMマニュアル	日立ソフトウェアマニュアルと製品群別CD-ROMマニュアルの仕様について記載しています。
マニュアルのご購入	マニュアルご購入時のお申し込み方法を記載しています。
オンラインマニュアル	一部製品のマニュアルをインターネットで公開しています。
サポートサービス	ソフトウェアサポートサービスお客様向けページでのマニュアル公開サービスを記載しています。
ご意見・お問い合わせ	マニュアルに関するご意見、ご要望をお寄せください。

## 2. インターネットでのマニュアル公開

2種類のマニュアル公開サービスを実施しています。

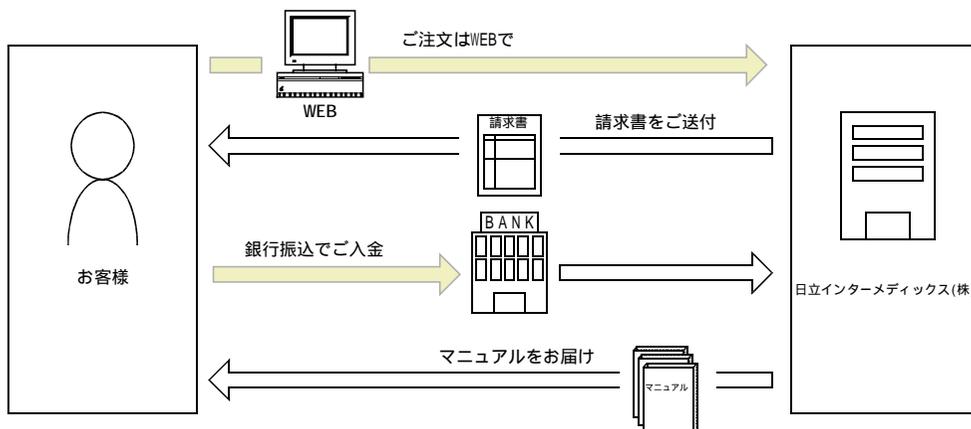
### (1) マニュアル情報ホームページ「オンラインマニュアル」での公開

製品をよりご理解いただくためのご参考として、一部製品のマニュアルを公開しています。

### (2) ソフトウェアサポートサービスお客様向けページでのマニュアル公開

ソフトウェアサポートサービスご契約のお客様向けにマニュアルを公開しています。公開しているマニュアルの一覧、本サービスの対象となる契約の種別などはマニュアル情報ホームページの「サポートサービス」をご参照ください。

## 3. マニュアルのご注文



マニュアル情報ホームページの「マニュアルのご購入」にアクセスし、お申し込み方法をご確認のうえWEBからご注文ください。ご注文先は日立インターメディアックス(株)となります。

ご注文いただいたマニュアルについて請求書をお送りします。

請求書の金額を指定銀行へ振り込んでください。

入金確認後7日以内にお届けします。在庫切れの場合は、納期を別途ご案内いたします。