



Solaris WBEM Services の管理

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

Part No: 816-3968-10
2002 年 5 月

Copyright 2002 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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

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

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

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

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

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

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

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

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

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

「ATOK」は、株式会社ジャストシステムの登録商標です。

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

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

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

Unicode は、Unicode, Inc. の商標です。

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

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

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

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

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

原典: Solaris WBEM Services Administration Guide

Part No: 806-6827-10

Revision A



020326@2851



目次

はじめに	11
1 WBEM と Solaris WBEM Services (概要)	15
Web-Based Enterprise Management について	15
Common Information Model について	16
基本的な CIM 要素	16
CIM モデル	17
CIM エクステンション (拡張)	17
Solaris WBEM Services	18
ソフトウェアのコンポーネント	19
ネームスペース	22
プロバイダ	23
ほかの WBEM システムとの相互運用性	24
Solaris WBEM Software Developer's Kit	24
2 CIM Object Manager の使用 (手順)	25
CIM Object Manager について	25
init.wbem コマンド	26
Solaris Management Console サーバー	27
システムのブート	27
CIM Object Manager の停止と再起動	27
▼ CIM Object Manager を停止する方法	27
▼ CIM Object Manager を再起動する方法	27
CIM Object Manager Repository のアップグレード (作業マップ)	28
CIM Object Manager Repository のアップグレード	28

▼ JavaSpaces データストアを保存する方法	29
▼ WBEM データを変換する方法	30
▼ WBEM データをマージする方法	31
例外メッセージ	32
3 WBEM 用 SNMP サービスの使用方法 (手順)	33
WBEM 用 SNMP アダプタ	33
WBEM 用 SNMP アダプタの動作のしくみ	34
マスターエージェントが要求を転送する方法: WBEM 用 SNMP アダプタと Sun SNMP Agent の比較	35
アダプタの構成と SNMP から CIM オブジェクトへのマッピング	35
構成ファイル	35
マッピングファイル	38
WBEM 用 SNMP アダプタのインストールと使用 (作業マップ)	43
WBEM 用 SNMP アダプタのインストールと使用	43
▼ WBEM 用 SNMP アダプタをインストールする方法	43
▼ WBEM 用 SNMP アダプタを起動する方法	44
▼ WBEM 用 SNMP アダプタを使用不可にする方法	44
▼ WBEM 用 SNMP アダプタにマッピングファイルディレクトリを再び強制的に 読み取らせる方法	45
WBEM 用 SNMP アダプタに関する問題の障害追跡	45
要求の送受信	46
FIFO を開くことができない	47
FIFO を作成できない	48
WBEM サービスが起動されていない	49
SNMP プロバイダ	50
SNMP MIB ファイルから MOF ファイルを生成する	50
▼ SNMP MIB ファイルから MOF ファイルを生成する方法	50
4 セキュリティの管理 (手順)	53
WBEM のセキュリティ機構	53
認証	54
役割の引き受け	54
セキュリティ保護されたメッセージング	55
承認	55
監査	57
ロギング	57

Sun WBEM User Manager を使ってアクセス制御を設定する	58
Sun WBEM User Manager で実行できることと実行できないこと	58
Sun WBEM User Manager の使用 (作業マップ)	59
Sun WBEM User Manager の使用方法	60
▼ Sun WBEM User Manager を起動する方法	60
▼ ユーザーにデフォルトのアクセス権を許可する方法	61
▼ ユーザーのアクセス権を変更する方法	61
▼ ユーザーのアクセス権を削除する方法	61
▼ ネームスペースのアクセス権を設定する方法	62
▼ ネームスペースのアクセス権を削除する方法	62
WBEM のセキュリティに関する問題の障害追跡	63
クライアント (ユーザー) が WBEM サーバー上の CIM Object Manager によって認 証されない場合	63
そのほかの CIM セキュリティ例外エラーが起きる場合	65
認証検査に失敗する	65
5 システムログデータの表示 (手順)	67
ログビューアを使ってログデータを調べる	67
▼ Solaris Management Console アプリケーションとログビューアを起動する方 法	67
索引	69

表目次

表 2-1	WBEM データを変換するかマージするかの判別	29
表 3-1	マッピングファイルの内容	41

図目次

- 図 1-1 Solaris WBEM Services のアーキテクチャ 19
- 図 5-1 Solaris Management Console アプリケーションで「ログビューア (Log Viewer)」を選択した状態 68

はじめに

『Solaris WBEM Services の管理』では、CIM (Common Information Model) の概念、および Solaris™ オペレーティング環境での WBEM (Web-Based Enterprise Management) サービスの管理方法について説明します。

Solaris WBEM Services ソフトウェアを使用することにより、ソフトウェア開発者は Solaris 上で稼働する管理アプリケーションをより簡単に作成できるようになり、また、Solaris オペレーティング環境をより簡単に管理できるようになります。

対象読者

このマニュアルは、既存の WBEM アプリケーションを使用したり、新しいアプリケーションを作成したりして、WBEM 対応のネットワークやシステムを管理する、システム管理者を対象にしています。

お読みになる前に

このマニュアルは、読者に次の知識があることを前提としています。

- オブジェクト指向プログラミングの概念
- Java™ プログラミング
- WBEM CIM (Common Information Model) の概念
- ネットワーク管理の概念
- SNMP (Simple Network Management Protocol) の概念 (WBEM の SNMP アダプタを構成し、使用する場合)

知識が不十分な場合には、次の書籍を参考にご覧を勧めします。

- 『Java How to Program』 H. M. Deitel, P. J. Deitel 共著、Prentice Hall、ISBN 0-13-263401-5
- 『The Java Class Libraries, Second Edition, Volume 1』 Patrick Chan, Rosanna Lee, Douglas Kramer 共著、Addison-Wesley、ISBN 0-201-31002-3
- Distributed Management Task Force の「CIM Tutorial」
(<http://www.dmtf.org/education/cimtutorial.php>)

次に、WBEM 技術に携わる場合に有用な Web サイトを示します。

- 「CIM Tutorial Glossary」 -
www.dmtf.org/education/cimtutorial/reference/glossary.php
- 「Distributed Management Task Force (DMTF)」 - www.dmtf.org
このサイトには、CIM の最新情報、各種の作業グループについての情報、および CIM スキーマの拡張についての情報が掲載されています。
- 「Rational Software」 - www.rational.com/uml
このサイトでは、Unified Modeling Language (UML) および Rose CASE ツールの関連文書を入手できます。

内容の紹介

第 1 章では、WBEM (Web-Based Enterprise Management) および Solaris WBEM Services の概要を説明しています。

第 2 章では、CIM とは何かを説明しています。CIM Object Manager の起動と停止方法、および CIM Object Manager Repository のアップグレードの方法について説明しています。

第 3 章では、WBEM 用 SNMP サービスについて説明します (WBEM および SNMP Provider 用 SNMP アダプタを含む)。WBEM 用 SNMP アダプタは、システム管理者による使用を前提としているため、SNMP (Simple Network Management Protocol) 管理アプリケーションで Solaris WBEM Service のシステム管理情報にアクセスすることを可能にします。SNMP プロバイダは、CIM Object Manager へ管理要素についての情報 (SNMP デバイスの構成情報など) を提供するソフトウェアコンポーネントです。

第 4 章では、WBEM セキュリティ機構、セキュリティ機能、およびネームスペースやユーザーへのアクセス権の設定方法を説明します。

第 5 章では、ログ情報の表示方法を説明します。

Sun のオンラインマニュアル

docs.sun.com では、Sun が提供しているオンラインマニュアルを参照することができます。マニュアルのタイトルや特定の主題などをキーワードとして、検索を行うこともできます。URL は、<http://docs.sun.com> です。

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を使用してすべてのファイルを表示します。 <code>system%</code>
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	<code>system% su</code> <code>password:</code>
<i>AaBbCc123</i>	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、 <code>rm filename</code> と入力します。
『 』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。
「 」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	<code>sun% grep '^#define \</code> <code>XV_VERSION_STRING'</code>

コード例は次のように表示されます。

- C シェル

```
machine_name% command y|n [filename]
```

- C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

- Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

- Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

一般規則

- このマニュアルでは、英語環境での画面イメージを使っています。このため、実際に日本語環境で表示される画面イメージとこのマニュアルで使っている画面イメージが異なる場合があります。本文中で画面イメージを説明する場合には、日本語のメニュー、ボタン名などの項目名と英語の項目名が、適宜併記されています。
- このマニュアルでは、「IA」という用語は、Intel 32 ビットのプロセッサアーキテクチャを意味します。これには、Pentium、Pentium Pro、Pentium II、Pentium II Xeon、Celeron、Pentium III、Pentium III Xeon の各プロセッサ、および AMD、Cyrix が提供する互換マイクロプロセッサチップが含まれます。

第 1 章

WBEM と Solaris WBEM Services (概要)

この章では、Web-Based Enterprise Management (WBEM) と Solaris WBEM Services の概要を説明します。これらのサービスを使用すれば、ソフトウェア開発者は Solaris で動作する管理アプリケーションをより簡単に作成できるようになり、システム管理者は Solaris オペレーティング環境をより容易に管理できるようになります。

この章の内容は以下のとおりです。

- 15 ページの「Web-Based Enterprise Management について」
- 16 ページの「Common Information Model について」
- 18 ページの「Solaris WBEM Services」
- 24 ページの「Solaris WBEM Software Developer's Kit」

Web-Based Enterprise Management について

業界全体をリードする取り組みである WBEM には、異機種プラットフォーム上のシステム、ネットワーク、およびデバイスを Web 上で管理するための標準規格が含まれています。この標準化により、システム管理者はデスクトップ、デバイス、およびネットワークを管理することができます。

今のところ、WBEM は、Simple Network Management Protocol (SNMP) と互換性があるように設計されています。

WBEM は、次の標準規格を包含しています。

- Common Information Model (CIM) – 管理リソースを記述するための情報モデル
- Managed Object Format (MOF) – CIM クラスとインスタンスを定義するための言語

- eXtensible Markup Language (XML) – Web 上の管理リソースを記述するためのマークアップ言語

コンピュータ業界とテレコミュニケーション業界の企業を代表するグループの1つである「Distributed Management Task Force (DMTF)」は、管理の標準規格の開発で主導的な立場にあります。DMTFの目的は、さまざまなプラットフォームおよびプロトコル上でネットワークを管理する統合的な手法を開発し、その結果、費用効率が高く、相互運用性に優れ、できるだけ欠陥のない製品を提供することにあります。

Common Information Model について

この節では、Solaris WBEM Services 製品で使用されている CIM の基本的な用語と概念を簡単に説明します。CIM の用語と概要の完全な用語集

は、<http://www.dmtf.org/education/cimtutorial/reference/glossary.php> にあります。

CIM とは、ディスク、CPU、オペレーティングシステムなどの管理リソースを記述するためのオブジェクト指向情報モデルです。CIM オブジェクトは、プリンタ、ディスクドライブ、CPU などの管理リソースを表すモデルです。CIM オブジェクトは、WBEM 対応のあらゆるシステムやデバイス、アプリケーションの間で共有できます。

基本的な CIM 要素

類似したプロパティや目的を持つ CIM オブジェクトは、CIM クラスで表現されます。プロパティは、クラスのデータユニットを記述する属性です。インスタンスは、特定のクラスの実際の管理オブジェクトを表したものです。インスタンスには実際のデータが含まれています。たとえば、Solaris ComputerSystem は、Solaris オペレーティング環境を実行するコンピュータを表す CIM クラスです。システム上で実行されている Solaris ソフトウェアは、Solaris OperatingSystem クラスのインスタンスです。ResetCapability や InstallDate は Solaris ComputerSystem クラスのプロパティの例です。

CIM クラスは、スキーマと呼ばれる意味のある集合にグループ分けされます。スキーマは、1つの所有者(または組織)に所属する、複数のクラスからなるグループです。クラスはただ1つのスキーマに所属しなければなりません。スキーマは、管理やクラスの名前付けに使用されます。同じスキーマ内ではすべてのクラス名が異ならなければなりません。クラスやプロパティの名前が同じである場合、スキーマ名はクラスやプロパティを区別する1つの要素となります。スキーマ、クラス、プロパティの名前は、次の構文に従って付けます。

Schemaname_classname.propertyname

CIM モデル

Common Information Model (CIM) は、情報を一般的なものから特定のものまで分類します。Solaris の環境などの特定の情報は、このモデルを拡張して記述されています。CIM は、次に示す 3 つの情報層から構成されます。

- コアモデル – プラットフォームに依存しない、CIM のサブセット
- 共通モデル – ネットワーク管理の特定の領域に関連するエンティティ (システム、デバイス、アプリケーションなど) の概念、機能性、および表示方法を視覚的に表す情報モデル
- エクステンション (拡張) – CIM スキーマをサポートし、限定されたプラットフォーム、プロトコル、または企業独自のものを表す情報モデル

コアモデルと共通モデルを、総称して CIM スキーマと呼びます。

コアモデル

コアモデルは、管理環境の基本となる一般的な前提事項を提供します。たとえば、要求された特定のデータは、ある場所に格納され、要求元のアプリケーションまたはユーザーに配布されなければならない、などの前提事項です。これらの前提事項は、管理環境の基盤を概念的に形成する、クラスと関連のセットとして示されます。コアモデルは、管理環境の特定の側面を表現するスキーマに一貫性を持たせます。

コアモデルは、クラス、関連、およびプロパティのセットをアプリケーション開発者に提供します。開発者は、このセットを使用して管理対象システムを表現し、共通モデルを拡張する方法を決定することができます。コアモデルは、その他の管理環境をモデル化する概念的な枠組みを確立します。

コアモデルは、共通モデルとエクステンション (拡張) が、システム、アプリケーション、ネットワーク、デバイスなどのネットワーク機能に関する特定の情報を拡張するためのクラスと関連を提供します。

共通モデル

共通モデルで示されるネットワーク管理の領域は、特定の技術や実装には依存せず、管理アプリケーションの開発基盤を提供します。このモデルは、指定された 5 つの技術別スキーマ、Systems、Devices、Applications、Networks、および Physical に、拡張用の基底クラスセットを提供します。

CIM エクステンション (拡張)

拡張スキーマは、このモデルに特定の技術に関連づけるために CIM に組み込まれます。CIM を拡張すると、より多数のユーザーと管理者が Solaris などの特定のオペレーティング環境を管理できるようになります。拡張スキーマのクラスを使用して、ソフトウェア開発者はこれらの技術を管理するアプリケーションを開発することができます。Solaris スキーマは、CIM スキーマを拡張したものです。

Solaris WBEM Services

Solaris WBEM Services は、Solaris オペレーティング環境で、管理データのセキュリティ保護されたアクセスと操作などの、WBEM サービスを提供するソフトウェアです。製品には Solaris プロバイダが組み込まれているため、管理アプリケーションから Solaris オペレーティング環境の管理リソース (デバイスやソフトウェア) の情報にアクセスできます。

CIM Object Manager は、Remote Method Invocation (RMI) プロトコルか XML/HTTP プロトコルを使用する管理アプリケーションからの接続を受け入れ、接続されたクライアントに次のようなサービスを提供します。

- 管理サービス – CIM Object Manager は、CIM データの意味と構文を検査し、アプリケーション、CIM Object Manager Repository、および管理リソースの間でデータの送受信を行います。
- セキュリティサービス – これらのサービスは、Solaris Management Console のユーザーツールによって WBEM に指定します。これらのサービスについては、『Solaris のシステム管理 (セキュリティサービス)』を参照してください。
- Sun™ WBEM User Manager – このツールを使用して、WBEM サーバー上の特定のネームスペースのアクセス制御リスト (ACL) を確立します。Sun WBEM User Manager では、承認されたユーザーの追加と削除、承認されたユーザーに対するアクセス特権の設定、および WBEM 対応システム上の CIM オブジェクトに対するユーザー認証とアクセスの管理を実行できます。ACL ベースのセキュリティは、Solaris WBEM Services によって提供される独自の機能です。
- ロギングサービス – このサービスを構成するクラスを使えば、開発者は、イベントデータを動的に記録したり取得したりできるアプリケーションを作成できます。管理者はこのデータを使ってイベントの原因を追跡したり、判定したりすることができます。ロギングサービスについての詳細は、『Solaris WBEM SDK 開発ガイド』を参照してください。
- XML サービス – XML データを CIM クラスに変換します。XML/HTTP ベースの WBEM クライアントが CIM Object Manager と通信できるようにします。

WBEM 対応システムに接続されると、WBEM クライアントは、次のような WBEM 操作を要求できます。すなわち、CIM クラスおよびインスタンスの作成、表示、削除や、指定された値をもつプロパティの検索、指定されたクラス階層にあるインスタンスやクラスの列挙 (リストの取得) などです。

ソフトウェアのコンポーネント

Solaris WBEM Services ソフトウェアは、アプリケーション、管理、およびプロバイダという 3 つの層で機能するソフトウェアコンポーネントで構成されます。これらのコンポーネントはオペレーティングシステムやハードウェアとデータを送受信します。次の図に、各ソフトウェアコンポーネントと、それぞれの送受信方法を示します。

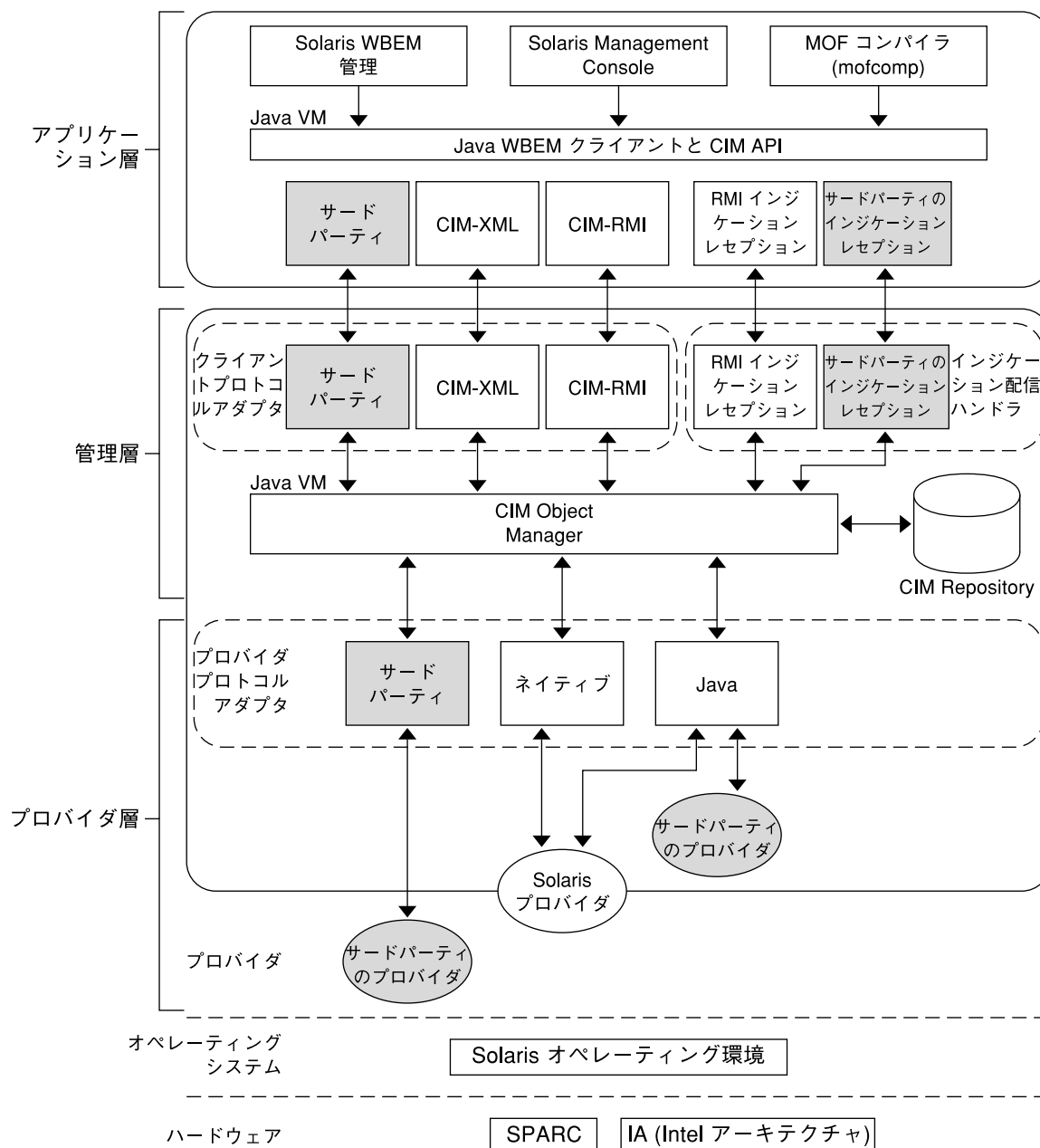


図 1-1 Solaris WBEM Services のアーキテクチャ

- アプリケーション層 - WBEM クライアントが管理リソースからのデータを処理したり、表示したりします。Solaris WBEM Services には、次のアプリケーションがあります。

- Sun WBEM User Manager および Solaris Management Console ユーザーツール – これらのアプリケーションでは、システム管理者が、承認されたユーザーの追加や削除を実行したり、それらのユーザーの管理リソースへのアクセス特権を設定したりできます。
- Solaris Management Console ログビューア – ログファイルを表示するアプリケーションです。ログに残ったコマンドを実行したユーザーの名前や、ログに残ったイベントが発生したクライアントコンピュータなど、ログレコードの詳細を表示できます。
- **Managed Object Format (MOF) コンパイラ** – このプログラムは、MOF 文を含むファイルを解析し、そのファイルで定義されているクラスやインスタンスを Java クラスに変換し、その Java クラスを CIM Object Manager Repository (管理データを一元的に格納する場所) に追加します。
MOF は、CIM のクラスやインスタンスを定義する言語です。MOF ファイルは、MOF 言語を使って CIM オブジェクトを記述する ASCII テキストファイルです。CIM オブジェクトは、プリンタ、ディスクドライブ、CPU などの管理リソースを表したモデルです。MOF ファイルは /usr/sadm/mof にあります。
管理リソースの情報は MOF ファイルに格納されることがあります。MOF は Java に変換できるため、Java 仮想マシン (JVM) を持つシステムで動作するアプリケーションならこの情報の解釈や交換を行うことができます。さらに、インストールの後で、mofcomp コマンドを使って MOF ファイルをいつでもコンパイルできます。MOF については、DMTF の Web ページ <http://www.dmtf.org> を参照してください。
- 管理層 – この層のコンポーネントは、接続された WBEM クライアントに次のサービスを提供します。
 - Common Information Model (CIM) Object Manager – WBEM システム上の CIM オブジェクトを管理するソフトウェアです。CIM オブジェクトは内部的には Java クラスとして格納されます。CIM Object Manager は WBEM クライアント、CIM Object Manager Repository、管理リソースとの間で情報を送受信します。
 - **CIM Object Manager Repository** – CIM のクラスやインスタンスの定義を一元的に格納する場所です。
 - クライアントおよび CIM アプリケーションプログラミングインタフェース (API) – WBEM クライアントアプリケーションは、これらの Java インタフェースを使って、管理リソースのクラスやインスタンスの作成および表示などの操作を CIM Object Manager に要求します。
 - プロバイダインタフェース – プロバイダは、これらのインタフェースを使って管理リソースの情報を CIM Object Manager に転送します。CIM Object Manager は、プロバイダインタフェースを使って、ローカルにインストールされたプロバイダに情報を転送します。
- プロバイダ層 – プロバイダは、CIM Object Manager と 1 つまたは複数の管理リソースとの間の仲介を行います。CIM Object Manager は、WBEM クライアントから、CIM Object Manager Repository に存在しないデータを要求されると、要求を適切なプロバイダに転送します。

- Solaris プロバイダ – Solaris オペレーティング環境内の管理リソースのインスタンスを、CIM Object Manager に提供します。プロバイダは、管理デバイスに関する情報の取得および設定を行います。ネイティブプロバイダとは、管理デバイスで動作するように作成されたマシン固有のプログラムです。たとえば、Solaris オペレーティング環境を実行しているシステム上のデータにアクセスするプロバイダには、そのシステムに照会する C 関数が含まれているはずで、Java Native Interface は、JDK™ ソフトウェアの一部です。Java Native Interface を使ってプログラムを作成すれば、その Java プログラムコードはどのプラットフォームに移植しても確実に動作します。Java Native Interface を使うと、Java 仮想マシン内で動作する Java コードを、C、C++、アセンブラなど、他の言語で作成されたアプリケーションやライブラリとともに動作することができます。
- Solaris スキーマ – Solaris オペレーティング環境内の管理対象オブジェクトを記述するクラスの集合です。CIM スキーマや Solaris スキーマのクラスは CIM Object Manager Repository に格納されます。CIM スキーマは、どの管理環境にもある管理オブジェクトを表すためのクラス定義の集合です。
Solaris スキーマは CIM スキーマを拡張したもので、一般的な Solaris オペレーティング環境の管理オブジェクトを表すクラス定義の集合です。ユーザーは、MOF コンパイラ (mofcomp) を使用して CIM スキーマ、Solaris スキーマ、あるいはその他のクラスを CIM Object Manager Repository に追加することもできます。
- オペレーティングシステム層 – Solaris プロバイダを使えば、管理アプリケーションから Solaris オペレーティング環境にある管理リソース (デバイスおよびソフトウェア) の情報にアクセスできます。
- ハードウェア層 – 管理クライアントは、サポートされる任意の Solaris プラットフォームの管理データにアクセスできます。

ネームスペース

ネームスペースと呼ばれるディレクトリのような構造には、1 つまたは複数のスキーマを格納できます。CIM ネームスペースには、他のネームスペース、クラス、インスタンス、および修飾子型を格納できます。1 つのネームスペース内のオブジェクト名は、一意でなければなりません。

Solaris WBEM Services では、WBEM クライアントアプリケーションが特定のネームスペースに接続されると、それ以後のすべての操作はそのネームスペース内で行われます。ネームスペースに接続されているクライアントは、そのネームスペースとそのネームスペースに含まれるすべてのネームスペースのクラスやインスタンスに (存在すれば) アクセスできます。たとえば、child というネームスペースを root\cimv2 ネームスペースに作成すれば、root\cimv2 に接続することにより、root\cimv2 と root\cimv2\child ネームスペースのクラスやインスタンスにアクセスできます。

アプリケーションを、あるネームスペース内のネームスペースに接続することができます。これはディレクトリ内のサブディレクトリを使用するのと同じことです。アプリケーションを新しいネームスペースに接続すると、それ以後のすべての操作はその

ネームスペース内で行われます。たとえば、アプリケーションを `root\cimv2\child` ネームスペースに接続すると、そのネームスペースのすべてのクラスやインスタンスにはアクセスできますが、親ネームスペース `root\cimv2` のクラスやインスタンスにはアクセスできません。

インストール時に、次のネームスペースがデフォルトで作成されます。

- `root` – 他のネームスペースを含む、一番上位にあるネームスペース
- `root\cimv2-LogicalDisk`、`Netcard` など、システム上のオブジェクトを表すデフォルトの CIM クラスおよびインスタンスが入っている。これがデフォルトのネームスペース
- `root\security` – CIM Object Manager が使用する、ユーザーやネームスペースのアクセス権を表すセキュリティクラスが入る
- `root\snmp` – WBEM 用の SNMP プロバイダおよび SNMP アダプタのためのクラスが入る
- `root\system` – CIM Object Manager の情報とプロバイダのパスが入る

プロバイダ

WBEM クライアントアプリケーションが CIM データにアクセスすると、WBEM システムは現在のホストにあるユーザーのログイン情報を検証します。ユーザーには、デフォルトで CIM スキーマと Solaris スキーマに対する読み取り権が与えられます。CIM スキーマは、システムにあるすべての管理オブジェクトを標準形式で記述します。WBEM 対応のすべてのシステムやアプリケーションは、この標準形式を解釈できます。

プロバイダとは、管理オブジェクトと通信してデータにアクセスするクラスです。プロバイダは、この情報を Repository に統合し解釈するために CIM Object Manager に転送します。CIM Object Manager Repository に存在しないデータの要求を管理アプリケーションから受け取る場合、CIM Object Manager はその要求をプロバイダに送ります。

CIM Object Manager は、オブジェクトプロバイダの API を使用してプロバイダと通信します。アプリケーションが動的データを CIM Object Manager に要求すると、CIM Object Manager はプロバイダインタフェースを使って要求をプロバイダに渡します。

プロバイダは、CIM Object Manager からの要求に回答して次の機能を実行します。

- 固有の情報形式を CIM クラスに変換します。
 - デバイスから情報を取得します。
 - この情報を CIM クラスの形式で CIM Object Manager に渡します。
- CIM クラスからの情報をデバイス固有の形式に変換します。
 - 必要な情報を CIM クラスから取得します。
 - 取得した情報を、デバイスに固有のデバイス形式で渡します。

ほかの WBEM システムとの相互運用性

WBEM クライアントと WBEM システムは、同じシステムで動作することも、別々のシステムで動作することもできます。WBEM システムには複数の WBEM クライアントを接続できます。たとえば、1 つの WBEM システムで 4 つまたは 5 つの WBEM クライアントをサポートすることが可能です。

Solaris WBEM Services は、Version 1.1 Specification for CIM Operations over HTTP をサポートします。この仕様では、XML を使って CIM のオブジェクトやメッセージを記述します。XML は、Web 上のデータを記述するための標準マークアップ言語です。この仕様では、XML マークアップを拡張して CIM のオブジェクトや操作を定義します。XML は Web 上で送信可能なデータを記述する標準的な方法であるため、WBEM クライアントは、XML データを解析できる任意の WBEM システム上の CIM データにアクセスできます。

Solaris WBEM Software Developer's Kit

Solaris WBEM Software Developer's Kit (SDK) には、WBEM 対応の管理デバイスと通信する管理アプリケーションを作成するのに必要なコンポーネントが含まれています。さらに、開発者は、この SDK を使ってプロバイダを作成できます。プロバイダとは、管理対象オブジェクトと通信してデータにアクセスするプログラムのことです。Solaris WBEM SDK を使って開発したすべての管理アプリケーションは、Java 環境で実行できます。

WBEM クライアントアプリケーションは、Solaris WBEM API を使って CIM オブジェクトを操作するプログラムです。クライアントアプリケーションは、通常、CIM API を使ってオブジェクト (ネームスペース、クラス、インスタンスなど) を構築し、初期化します。次に、クライアントアプリケーションはクライアント API を使ってオブジェクトを CIM Object Manager に渡し、CIM のネームスペース、クラス、インスタンスの作成などの WBEM 操作を要求します。

Solaris WBEM SDK は、Java 環境にインストールして、使用できます。Solaris WBEM SDK は、スタンドアロンのアプリケーションとして使用することも、Solaris WBEM Services とともに使用することもできます。

Solaris WBEM SDK については、『Solaris WBEM SDK 開発ガイド』を参照してください。

第 2 章

CIM Object Manager の使用 (手順)

Common Information Model (CIM) Object Manager は、WBEM クライアントアプリケーションと管理リソースとの間で CIM データを送受信するソフトウェアです。

この章の内容は以下のとおりです。

- 28 ページの「CIM Object Manager Repository のアップグレード (作業マップ)」
- 25 ページの「CIM Object Manager について」
- 26 ページの「init.wbem コマンド」
- 27 ページの「CIM Object Manager の停止と再起動」
- 32 ページの「例外メッセージ」

CIM Object Manager について

CIM Object Manager は、WBEM 対応のシステム上の CIM オブジェクトを管理します。CIM オブジェクトは、プリンタ、ディスクドライブ、CPU などの管理リソースを表したモデルです。CIM オブジェクトは内部的には Java クラスとして格納されます。

WBEM クライアントアプリケーションが CIM オブジェクトについての情報にアクセスすると、CIM Object Manager は、そのオブジェクトに適したプロバイダか、CIM Object Manager Repository と接続します。プロバイダとは、管理オブジェクトと通信してデータにアクセスするクラスです。WBEM クライアントアプリケーションが CIM Object Manager Repository に存在しない管理リソースのデータを要求すると、CIM Object Manager は、その管理リソースのプロバイダに要求を転送します。プロバイダは情報を動的に取得します。

CIM Object Manager は、起動時に次の機能を実行します。

- RMI ポート 5987 上の RMI 接続と、HTTP ポート 5988 上の XML/HTTP 接続を待機する

- CIM Object Manager Repository との接続を設定する
- 要求が入ってくるのを待つ

CIM Object Manager は、次のことを行います。

- セキュリティ検査を実行して、ユーザーログインを認証し、ネームスペースへのアクセスを承認する
- CIM データ操作の構文と意味を検査して、最新の CIM 仕様に準拠していることを確認する
- 適切なプロバイダか CIM Object Manager Repository に要求を転送する
- プロバイダや CIM Object Manager Repository から受け取ったデータを WBEM クライアントアプリケーションに転送する

WBEM クライアントアプリケーションは、WBEM 操作 (CIM クラスの作成や CIM インスタンスの更新など) を実行する必要がある場合、CIM Object Manager にアクセスして接続を確立します。WBEM クライアントアプリケーションは、CIM Object Manager に接続されると、CIM Object Manager への参照を取得し、この参照を使ってサービスや操作を要求します。

init.wbem コマンド

Solaris は、インストール時、およびシステムのリブートのたびに、自動的に `init.wbem` を実行します。`init.wbem` コマンドは、CIM Object Manager と Solaris Management Console サーバーを起動し、これらは連動して単一プロセスで実行します。また、CIM Object Manager を停止したり、Solaris Management Console サーバーを停止したり、サーバーからステータスを入手したりするためにも、`init.wbem` コマンドを使用できます。このコマンドに関する付加的な情報は、`init.wbem(1M)` のマニュアルページにあります。

一般には CIM Object Manager を停止する必要はありません。ただし、既存のプロバイダを変更する場合には、変更後のプロバイダを使用する前に、CIM Object Manager を停止してから再起動する必要があります。

`init.wbem` コマンドには、次の 3 つのオプションを指定できます。

- `start` - ローカルホスト上の CIM Object Manager または Solaris Management Console サーバーを起動する
- `stop` - ローカルホスト上の CIM Object Manager および Solaris Management Console サーバーを停止する
- `status` - ローカルホスト上の CIM Object Manager および Solaris Management Console サーバーのステータスを取得する

Solaris Management Console サーバー

Solaris Management Console ソフトウェアは、ユーザー管理、ディスク管理、ログビューアなどの Solaris 管理アプリケーションを提供します。Solaris Management Console サーバーは、コンソールにダウンロードのためのツールを提供し、またコンソールおよびそのツールのために認証、承認、ロギング、メッセージング、および保持などの共通サービスを実行します。

Solaris Management Console については、このマニュアルの別の章で説明されており、『Solaris のシステム管理 (基本編)』にも説明があります。

システムのブート

`init.wbem` コマンドは、`/etc/init.d` ディレクトリにあります。`init state 2` に入った時点 (通常はブート時) で、`/etc/rc2.d/S90wbem` ファイルが `start` オプションを指定して実行されます。また、`init state 0, 1, および S` に入った時点 (通常は、システムの停止時か、システムがシステム管理者モードまたはシングルユーザーモードに入ったとき) で、`/etc/rc0.d/K36wbem` ファイル、`/etc/rc1.d/K36wbem` ファイル、および `/etc/rcS.d/K36wbem` ファイルが `stop` オプションを指定して実行されます。

CIM Object Manager の停止と再起動

プロバイダを変更する場合は、変更後のプロバイダを使用する前に、CIM Object Manager を停止し、再起動しなければなりません。

▼ CIM Object Manager を停止する方法

1. スーパーユーザーになります。
2. CIM Object Manager を停止します。

```
# /etc/init.d/init.wbem stop
```

▼ CIM Object Manager を再起動する方法

1. スーパーユーザーになります。
2. CIM Object Manager を再起動します。

```
# /etc/init.d/init.wbem start
```

CIM Object Manager Repository のアップグレード (作業マップ)

次の表は、CIM Object Manager Repository をアップグレードする手順を示しています。JavaSpaces™ データストアを保存して変換するか、WBEM データをマージするかは、Solaris 9 オペレーティング環境にアップグレードする前に使用していた Solaris のバージョンによって決まります。表 2-1 に、CIM Object Manager Repository をアップグレードする手順を決定する方法が説明されています。

作業	説明	参照先
JavaSpaces データストアを保存する	ファイルをダウンロードするかコピーすることによって JavaSpaces データストアを保存し、システムに現在インストールされている JDK のバージョンを判別する	29 ページの「JavaSpaces データストアを保存する方法」
WBEM データを変換する	wbemconfig convert コマンドを使用して、WBEM データを変換する	30 ページの「WBEM データを変換する方法」
WBEM データをマージする	wbemconfig convert コマンドを使用して、WBEM データをマージする	31 ページの「WBEM データをマージする方法」

CIM Object Manager Repository のアップグレード

独自にカスタマイズした Managed Object Format (MOF) データは、Solaris 9 の WBEM Services 2.5 で使用される新しい Reliable Log Repository 書式に更新しなければなりません。

Solaris 9 オペレーティング環境にアップグレードする前に、JavaSpaces データストアの保存が必要になる場合があります。また、Solaris 9 にアップグレードしたあと、アップグレードする前に使用していた Solaris のバージョンに応じて、データを変換するか、またはマージしなければなりません。

データの変換またはマージ処理を正しく実行しないと、データが損失します。

次の表を使って、アップグレードする前に JavaSpaces データストアを保存するかどうか、および Solaris 9 オペレーティング環境にアップグレードしたあとに WBEM データを変換するか、あるいはマージするかを判別してください。

表 2-1 WBEM データを変換するかマージするかの判別

Solaris 9 にアップグレードする前のオペレーティング環境	アップグレードする前に JavaSpaces データストアを保存するか	変換かマージか
Solaris 8 (Solaris WBEM Services 2.0)		
Solaris 8 6/00 (WBEM Services 2.0)	はい	変換
Solaris 8 10/00 (WBEM Services 2.2)		
Solaris 8 1/01 (WBEM Services 2.3)		
Solaris 8 4/01 (WBEM Services 2.4)		
Solaris 8 7/01 (WBEM Services 2.4)	いいえ	マージ
Solaris 8 10/01 (WBEM Services 2.4)		
Solaris 9 (Beta) (WBEM Services 2.5)		

▼ JavaSpaces データストアを保存する方法

1. スーパーユーザーになります。
2. 必要になるファイルをダウンロードするか、または現在の **JavaSpaces** データストアを保存するかを決めます。

注 - より安全な方法は、ファイルをダウンロードするよりも、現在の JavaSpaces データストアを保存することです。

- ファイルをダウンロードすることに決めた場合は、30 ページの「WBEM データを変換する方法」に進みます。
- 現在の JavaSpaces データストアを保存することに決めた場合は、次のコマンドを入力します。

```
# cd /usr/sadm/lib/wbem
# cp outrigger.jar outrigger.jar.tmp
# cp outrigger-dl.jar outrigger-dl.jar.tmp
# cp transient-outrigger.jar transient-outrigger.jar.tmp
# cp jini-core.jar jini-core.jar.tmp
# cp jini-ext.jar jini-ext.jar.tmp
# cp tools.jar tools.jar.tmp
# cp pro.zip pro.zip.tmp
```

3. システムに現在インストールされている **JDK** のバージョンを判別して記録しておきます。

```
# /usr/bin/java -version
java version "1.2.1"
Solaris VM (build Solaris_JDK_1.2.1_04c, native threads, sunwjit)
```

注 - WBEM データを変換するには、元の JavaSpaces データストアを作成したときに使用していた JDK と同じバージョンを実行していなければなりません。

▼ WBEM データを変換する方法

1. システムを **Solaris 9** オペレーティング環境にアップグレードします。
2. スーパーユーザーになります。
3. **CIM Object Manager** を停止します。

```
# /etc/init.d/init.wbem stop
```



注意 - `wbemconfig convert` コマンドを実行する前に **CIM Object Manager** を停止しなかった場合、データが損傷を受ける場合があります。

4. 29 ページの「**JavaSpaces** データストアを保存する方法」の手順で、現在の **JavaSpaces** データストアを保存したかどうかを確認します。
 - 保存した場合は、その **JavaSpaces** データストアを復元します。

```
# cd /usr/sadm/lib/wbem
# cp outrigger.jar.tmp outrigger.jar
# cp outrigger-dl.jar.tmp outrigger-dl.jar
# cp transient-outrigger.jar.tmp transient-outrigger.jar
# cp jini-core.jar.tmp jini-core.jar
# cp jini-ext.jar.tmp jini-ext.jar
# cp tools.jar.tmp tools.jar
# cp pro.zip.tmp pro.zip
```
 - 保存しなかった場合は、<http://www.sun.com/solaris/wbem> から `UpgradeRepository.zip` ファイルをダウンロードして解凍します。
`UpgradeRepository.zip` には、あとで WBEM データを変換するときに必要な `.jar` ファイルが含まれています。
5. 現在使用中の **JDK** がインストールされているディレクトリ以外の場所に、29 ページの「**JavaSpaces** データストアを保存する方法」で記録したバージョンの **JDK** を入手してインストールします。
6. シンボリックリンクを、`/usr/java` に現在インストールされている **JDK** から、29 ページの「**JavaSpaces** データストアを保存する方法」で記録したバージョンの **JDK** に変更します。

たとえば、現在インストールされている JDK を、/old_sdk 内の Solaris_JDK_1.2.1_04c に変更するには、次のように入力します。

```
# rm /usr/java
# ln -s /old_sdk/Solaris_JDK_1.2.1_04c /usr/java
```

7. **JavaSpaces** データストア内のデータを、**Reliable Log** 書式に変換します。

```
# /usr/sadm/lib/wbem/wbemconfig convert
```

wbemconfig convert コマンドでは、独自にカスタマイズした MOF データは正しく変換できますが、変更を加えた CIM または Solaris MOF データは変換できません。変更を加えた CIM または Solaris MOF データは壊れます。

注 – 変更を加えた CIM または Solaris MOF データを新しいレポジトリに再コンパイルするには、mofcomp コマンドを使用して、クラス定義を含む MOF ファイルをコンパイルします。

8. /usr/java からのシンボリックリンクを、**Solaris 9** オペレーティング環境とともに出荷された **JDK** ソフトウェアの場所に変更します。

たとえば、シンボリックリンクを /usr/java1.4 に変更するには、次のように入力します。

```
# rm /usr/java
# ln -s /usr/java1.4 /usr/java
```

9. **CIM Object Manager** を停止します。

```
# /etc/init.d/init.wbem stop
```

10. **CIM Object Manager** を起動します。

```
# /etc/init.d/init.wbem start
```

CIM Object Manager は、変換されたデータの入ったレポジトリファイルを、/var/sadm/wbem/logr/ ディレクトリに追加します。このディレクトリは、システムを Solaris 9 にアップグレードしたときに Solaris インストールプログラムが作成したものです。

▼ **WBEM** データをマージする方法

1. システムを **Solaris 9** オペレーティング環境にアップグレードします。
2. スーパーユーザーになります。
3. **CIM Object Manager** を停止します。

```
# /etc/init.d/init.wbem stop
```



注意 - `wbemconfig convert` コマンドを実行する前に CIM Object Manager を停止しなかった場合、データが損傷を受ける場合があります。

4. 以前のバージョンの **Reliable Log** にある元のデータを、**Solaris 9** の **Reliable Log** 内のデータとマージします。

```
# /usr/sadm/lib/wbem/wbemconfig convert
```

注 - `wbemconfig convert` コマンドでは、独自にカスタマイズした MOF データは正しく変換できますが、変更を加えた CIM または Solaris MOF データは変換できません。変更を加えた CIM または Solaris MOF データは壊れます。変更を加えた CIM または Solaris MOF データを新しいレポジトリに再コンパイルするには、`mofcomp` コマンドを使用して、クラス定義を含む MOF ファイルをコンパイルします。

例外メッセージ

CIM Object Manager は、MOF の構文や意味が正しくない場合、例外メッセージを生成します。例外メッセージについては、『*Solaris WBEM SDK 開発ガイド*』を参照してください。

第 3 章

WBEM 用 SNMP サービスの使用方法 (手順)

WBEM 用 SNMP サービスには、WBEM 用 SNMP アダプタおよび SNMP プロバイダという 2 つの要素があります。

この章の内容は以下のとおりです。

- 43 ページの「WBEM 用 SNMP アダプタのインストールと使用 (作業マップ)」
- 33 ページの「WBEM 用 SNMP アダプタ」
- 34 ページの「WBEM 用 SNMP アダプタの動作のしくみ」
- 35 ページの「アダプタの構成と SNMP から CIM オブジェクトへのマッピング」
- 45 ページの「WBEM 用 SNMP アダプタに関する問題の障害追跡」
- 50 ページの「SNMP プロバイダ」

WBEM 用 SNMP アダプタ

WBEM 用 SNMP アダプタは、システム管理者が使用することを意図して設計されたもので、Simple Network Management Protocol (SNMP) の管理アプリケーションが、Solaris WBEM Services によって提供されるシステム管理情報にアクセスすることを可能にします。

WBEM 用 SNMP アダプタは、snmpdx (Solstice Enterprise Agents™ 技術のマスターエージェント) とともに使用して、SNMP 要求を、それと同等の WBEM の Common Information Model (CIM) プロパティまたはインスタンスに対応づけます。マスターエージェントについては、snmpdx (1M) のマニュアルページを参照してください。

また、WBEM 用 SNMP アダプタは、CIM Object Manager からの応答を SNMP 応答に再び対応付けて、それを管理アプリケーションに返します。

マッピングファイルには、対応する Object Identifier (OID)、クラス名、プロパティ名、Abstract Syntax Notation One (ASN.1) タイプが含まれます。ユーザーは独自のマッピングファイルを作成できます。

WBEM 用 SNMP アダプタの動作のしくみ

Solaris オペレーティング環境は、Solstice Enterprise Agents のマスターエージェントを起動する前に、WBEM サービスを初期化します。デフォルトでは、WBEM 用 SNMP アダプタ (以下、アダプタ) は無効にされています。しかし、アダプタを有効にしたあとは、Solstice Enterprise Agents のマスターエージェント (snmpdx) がアダプタを自動的に起動します。アダプタについては、snmpXwbemd (1M) のマニュアルページに説明があります。

SNMP Manager は、SNMP の Get 要求を Solstice Enterprise Agents のマスターエージェントに渡します。マスターエージェントは、そのあと、Get 要求をアダプタに送り、アダプタは /var/sadm/wbem/snmp/map にあるマッピングファイルを使って、Get 要求内のオブジェクトを対応する CIM オブジェクトに変換します。さらに、アダプタは、CIM オブジェクトを Get 要求内の SNMP オブジェクトに変換します。

注 - 現在のところ、Solaris 9 では、Get 要求とスカラーオブジェクトだけがサポートされています。Get-next 要求、Get-bulk 要求、Set 要求、およびその他のオブジェクトは、現在のところサポートされていません。

アダプタは、このディレクトリを英数字順に検索して、拡張子 .map が付いている最初のファイルを見つけます。そのあと、アダプタは、ディレクトリ内のすべてのマッピングファイルを読み取り、その内容をキャッシュに書き込みます。アダプタは、これらのファイルの内容を使って、Get 要求の中に指定されたオブジェクトを、対応する CIM オブジェクトに変換します。アダプタは、ディレクトリ内の後続のマッピングファイルにある重複した OID を無視します。たとえば、次の OID が 002SUNWlvma.map にある場合、

```
1.3.6.1.2.1.1.1.0 My_ComputerSystem Description SnmpString
```

それと同じ次の OID が 050SUNWwbcou.map (アダプタはこのファイルを 002SUNWlvma.map よりあとに読み取る) にもあると、

```
1.3.6.1.2.1.1.1.0 Solaris_ComputerSystem Description SnmpString
```

アダプタは、050SUNWwbcou.map に指定された OID を無視します。

次いで、アダプタは、SNMP Manager が送信した各 Get 要求に対して、Get 応答を生成します。アダプタは、マッピングファイル内に対応する項目を見つけれない場合、Get 応答エラーを返します。

マスターエージェントが要求を転送する方法: WBEM 用 SNMP アダプタと Sun SNMP Agent の 比較

WBEM 用 SNMP アダプタがリリースされるまでは、SNMP Manager が SNMP MIB-2 変数のための Get 要求を Solstice Enterprise Agents のマスターエージェントに送信すると、マスターエージェントはその要求を Sun SNMP MIB-2 エージェント (mibiisa) に転送していました。アダプタが SNMP MIB-2 要求も処理することから考えると、Sun SNMP エージェントと WBEM 用 SNMP アダプタが両方同時に実行されている場合には、どうなるのでしょうか。マスターエージェントは要求をどのように転送するのでしょうか。

マスターエージェントは、各サブエージェント登録ファイルで定義されているサブツリーに基づいてノードテーブルを構築します。mibiisa サブエージェントは、MIB-2 サブツリー全体と、Sun Microsystems MIB サブツリーを登録します。アダプタは、MIB-2.system サブツリーと hostRsrc サブツリーを登録します。マスターエージェントは、2つのエージェントが同じサブツリーを登録することを許可しません。

Sun SNMP MIB-2 エージェントについては、mibiisa (1M) のマニュアルページを参照してください。マスターエージェントについては、『Solstice Enterprise Agents 1.0 ユーザーズガイド』を参照してください。

初期化の際に、マスターエージェントは、登録されている各サブツリーを含むノードテーブルを作成します。マスターエージェントは、各 Get 要求を、その要求に含まれている OID が最も適しているサブツリーのエージェントに転送します。たとえば、mib-2.system.5.0 に対する要求は、アダプタに転送されます。一方、mib-2.interfaces.1.0 に対する要求は、mibiisa サブエージェントに転送されます。マスターエージェントによって登録されたどのサブツリー内にも定義されていない OID の場合は、マスターエージェントは Get 応答でエラーを返します。

WBEM 用 SNMP アダプタは、SNMP V1 の要求だけをサポートしています。

アダプタの構成と SNMP から CIM オブジェクトへのマッピング

構成ファイル

この節では、WBEM 用 SNMP アダプタを構成するために使用するファイルについて説明します。そのファイルは、`/etc/snmp/conf/` にあります。

snmpXwbem.acl では、アダプタに関連付けられるアクセス制御リストの方針を、次の書式で定義します。

```
#pragma ident "@(#)snmpXwbem.acl 1.2 01/04/18 SMI"
#Copyright (c) 2001 by Sun Microsystems, Inc.
#All rights reserved.
```

```
# WBEM の SNMP サブエージェントの構成ファイル
```

```
#####
```

```
# アクセス制御 #
```

```
#####
```

```
# MIB 全体に対する読み取りおよび書き込みアクセスに必要な
```

```
# コミュニティ名のリスト
```

```
# リストが空の場合、有効なコミュニティ名は "public" だけで、
```

```
# そのアクセスタイプは読み取り専用
```

```
#
```

```
# managers リストに * がある場合は、任意のホストから
```

```
# 要求を受け取り可能であることを示している
```

```
acl = {
    {
        communities = public, private
        access = read-only
        managers = *
    }
}
```

```
#####
```

```
# トラップパラメータ #
```

```
#####
```

```
trap = {
}
```

複数のコミュニティおよびマネージャをコンマで区切って指定できます。アクセス方針は読み取り専用です。空の trap 句が必要です。Solaris 9 では、アダプタによってトラップはサポートされません。

snmpXwbem.reg では、アダプタが担当するサブツリーのオブジェクト識別子 (OID) を、次の書式で定義します。

```
#pragma ident "@(#)snmpXwbem.reg 1.3 01/10/04 SMI"
#
#Copyright (c) 2001 by Sun Microsystems, Inc.
#All rights reserved.
```

```
# WBEM の SNMP サブエージェントの構成ファイル
```

```
#####
```

```
# マクロ #
```

```
#####
```

```
# 次の 3 つのマクロが事前に定義されている
```

```

#
#   mib-2 =          1.3.6.1.2.1
#   enterprise =    1.3.6.1.4.1
#   sun =           1.3.6.1.4.1.42
#
# 独自のマクロを定義すれば、エージェントを定義するときに、
#  OID ではなく文字列を操作できる
# このあとのエージェントセクションを参照

macros = {
    system = mib-2.1
    hostRsrc = mib-2.25
}

#####
# エージェント#
#####

# 次のフィールドは必ず指定する必要がある
#
# - name:                エージェントの名前 (たとえば、エージェントの
#                        実行可能ファイルの名前)
#
# - subtrees:           エージェントがサポートしている OID / OID のサブツリーの
#                        リスト。リストに含める項目はコンマで区切らなければならない
#
#
# さらに、次のフィールドを変更または追加できる
#
# - timeout:            SNMP Relay がエージェントからの応答を待機する
#                        マイクロ秒数
#
# - watch-dog-time:     サブエージェントがアクティブかどうかをテストするために
#                        SNMP Relay が待機する秒数。watch-dog-time に指定された秒数
#                        の間、何かの活動があったかどうかテストされる
#
# - port:               エージェントを起動する UDP ポート番号
#
agents =
{
    {
        name = "WBEMsubagent"
        subtrees = { system, hostRsrc }
        timeout = 20000000
        watch-dog-time = 240
    }
}

```

timeout の単位はマイクロ秒です。watch-dog-time の単位は秒です。デフォルトでは、マスターエージェントは4分(または、watch-dog-time に設定された秒数)ごとにアダプタを起動しようとします。

注 - マスターエージェントは、アダプタによって使用されるポートを自動的に判別します。

snmpXwbem.rsrc- では、登録ファイルへのポインタを定義し、SNMP マスターエージェントがアダプタをどのように起動するかを次のような書式で定義します。

```
#pragma ident    "@(#)snmpXwbem.rsrc-    1.2    01/04/18 SMI"
#Copyright (c) 2001 by Sun Microsystems, Inc.
#All rights reserved.

#           WBEM の SNMP サブエージェント構成ファイル

#####
# エージェント #
#####
resource =
{
    {
        registration_file = "/etc/snmp/conf/snmpXwbem.reg"
        security = "/etc/snmp/conf/snmpXwbem.acl"
        policy = "spawn"
        type = "legacy"
        command = "/usr/sadm/lib/wbem/snmpXwbemd -p $PORT"
    }
}
}
```

マッピングファイル

マスターエージェントが Get 要求を WBEM 用 SNMP アダプタに渡すと、アダプタは、/var/sadm/wbem/snmp/map にあるマッピングファイルを使用して、その Get 要求を CIM オブジェクト要求に変換します。Solaris オペレーティング環境では、マッピングファイルがこのディレクトリに組み込まれています。さらに、SNMP Manager を使用して表示したい CIM 計測について、独自のマッピングファイルを定義することもできます。

この節では、Solaris オペレーティング環境が提供しているマッピングファイルの内容を紹介し、アダプタのマッピングファイルを作成するために知っておく必要のある事項を説明します。

Solaris に組み込まれているマッピングファイルの内容

この例は、Solaris オペレーティング環境に組み込まれているマッピングファイルの内容を示しています。

```
#
#pragma ident    "@(#)050SUNWwbcou.map    1.0    01/04/03 SMI"
#
```

```

# Copyright (c) 2001 by Sun Microsystems, Inc.
# All rights reserved.
#
# *** 内容の説明 ***
#
# コメント行とブランク行以外の最初の行は、必須のバージョンラベルを含む
# それ以降のコメント行とブランク行以外の行は、マッピング項目とみなされ、このあとの
# 説明に従って利用される
#
# 列 1 - SNMP OID - SNMP 変数を一意に記述する
# 例 2 - CIM クラス名 - この変数に関連付けられた CIM クラス
# 列 3 - CIM プロパティ名 - SNMP OID 変数に対応する CIM プロパティ
# 列 4 - ASN.1 型 - SNMP 要求との間でデータがどのように対応付けられるかを示す
#           SNMP データ型。サポートされている型: SnmpString、SnmpOid、
#           SnmpTimeticks、SnmpCounter、SnmpInt、SnmpGauge、SnmpIpAddress、
#           SnmpOpaque
# 列 5 以降は無視される
#
Version 1.0

1.3.6.1.2.1.1.1.0 Solaris_ComputerSystem Description SnmpString
1.3.6.1.2.1.1.3.0 Solaris_OperatingSystem LastBootUpTime SnmpTimeticks
1.3.6.1.2.1.1.4.0 Solaris_ComputerSystem PrimaryOwnerContact SnmpString
1.3.6.1.2.1.1.5.0 Solaris_ComputerSystem Name SnmpString

1.3.6.1.2.1.25.1.5.0 Solaris_OperatingSystem NumberOfUsers SnmpGauge
1.3.6.1.2.1.25.1.6.0 Solaris_OperatingSystem NumberOfProcesses SnmpGauge
1.3.6.1.2.1.25.1.7.0 Solaris_OperatingSystem MaxNumberOfProcesses SnmpGauge
1.3.6.1.2.1.25.1.2.0 Solaris_OperatingSystem LocalDateTime SnmpString

```

このマッピングファイルの内容は、SNMP MIB-2 システムグループのスカラーオブジェクトを、それぞれ対応する CIM オブジェクトに関連付けています。

MIB-2 システムグループのスカラーオブジェクト	CIM オブジェクト
sysDescr	Solaris_ComputerSystem.Description
sysUpTime	Solaris_OperatingSystem.LastBootUpTime
sysContact	Solaris_ComputerSystem.PrimaryOwnerContact
sysName	Solaris_ComputerSystem.Name

さらに、このマッピングファイルの内容は、SNMP Host Resources MIB オブジェクトを、それぞれ対応する CIM オブジェクトに関連付けています。

SNMP Host Resources MIB オブジェクト	CIM オブジェクト
hrSystemNumUsers	Solaris_OperatingSystem.NumberOfUsers

SNMP Host Resources MIB オブジェクト	CIM オブジェクト
hrSystemProcesses	Solaris_OperatingSystem.NumberOfProcesses
hrSystemMaxProcesses	Solaris_OperatingSystem.MaxNumberOfProcesses
hrSystemDate	Solaris_OperatingSystem.LocalDateTime

マッピングファイルの内容の構文については、40 ページの「マッピングファイルの内容の構文」で説明します。

注 - 現在のところ、ホストリソースデータを取得する唯一の方法は、CIM Object Manager を使用することです。これは、Solaris が現在、SNMP Host Resource エージェントを提供していないためです。

マッピングファイル名の構文

アダプタが確実にマッピングファイルを読み取るようにするには、次の構文に従ってファイル名を付けます。

alphanumeric-string.map

alphanumeric-string は、英数字の文字列を表します。たとえば、Solaris が組み込んであるマッピングファイルの名前は、次のとおりです。

050SUNWwbcou.map

アダプタが、より正確な順序でファイルを読み取るようにするため、先頭に 3 桁の数字を付けます。たとえば、002SUNWlvma.map は 050SUNWwbcou.map より前に読み取られます。

注 - 少なくとも root は、独自に作成したマッピングファイルを読み取ることができるようにする必要があります。

```
$ chmod 400 002SUNWlvma.map
```

マッピングファイルの内容の構文

次の表に、マッピングファイルの内容の要素と構文を示します。

表 3-1 マッピングファイルの内容

要素	説明	必須か
#	コメント。いつも無視される	いいえ
Version 1.0	マッピングファイルのバージョン。バージョンを示すテキスト文字列を、コメント以外の最初の行に指定しなければならない。このとおりにバージョンを指定しない場合、そのマッピングファイルは無視される	はい
1.3.6.1.2.1.1.1.0	SNMP オブジェクト識別子 (OID) で、SNMP 要求から抽出したいキー。SNMP OID は、SNMP 変数を記述している。アダプタは現在、スカラーだけをサポートしているため、OID は必ずテキスト文字列 .0 で終わる	はい
Solaris_ComputerSystem	変数に関連付けられる CIM クラス名	はい
Description	指定されたクラスの特徴を定義し、SNMP OID 変数に対応付ける、CIM プロパティ名	はい

表 3-1 マッピングファイルの内容 (続き)

要素	説明	必須か
SnmpString	<p>ASN.1 データ型。指定できる値と、その指定による対応付けは、次のとおり:</p> <ul style="list-style-type: none"> ■ SnmpString – 文字列、数値または CIM の LocalDateTime を SnmpString に移動する ■ SnmpInt – CIM の数値データ型 (数値形式の文字列を含む) を SnmpInt (符号付き、32 ビット整数) に移動する ■ SnmpCounter – CIM の数値データ型 (数値形式の文字列を含む) を SnmpCounter (符号なし、32 ビット整数) に移動する ■ SnmpGauge – CIM の数値データ型 (数値形式の文字列を含む) を SnmpGauge (符号なし、32 ビット整数) に移動する ■ SnmpTimeticks – 時刻の差 (100 分の 1 秒単位で表現) を SnmpTimeticks に移動する。この値は、CIM の値を現在の時刻から減算することによって計算される。たとえば、sysUpTime は currentTime から bootTime を減算することによって計算される ■ SnmpIpAddress – 文字列を SnmpIpAddress に移動する。文字列は、IP アドレス形式で指定しなければならない ■ SnmpOid – 文字列を SnmpOid に移動する。文字列は、OID 形式で指定しなければならない ■ SnmpOpaque – バイトのベクトルを SnmpOpaque に移動する 	はい

WBEM 用 SNMP アダプタのインストールと使用 (作業マップ)

次の表は、WBEM 用 SNMP アダプタ (以降、アダプタ) をインストール、起動、停止、および使用するために従う必要のある手順を示しています。

作業	説明	参照先
WBEM 用 SNMP アダプタをインストールする	Solaris オペレーティング環境をインストールする時に、アダプタをインストールする	43 ページの「WBEM 用 SNMP アダプタをインストールする方法」
WBEM 用 SNMP アダプタを起動する	snmpXwbem.rsrc- を snmpXwbem.rsrc に移動することによって WBEM 用 SNMP アダプタを起動する	44 ページの「WBEM 用 SNMP アダプタを起動する方法」
WBEM 用 SNMP アダプタを無効にする、または停止する	pkill コマンドを使用することによって、WBEM 用 SNMP アダプタを停止する	44 ページの「WBEM 用 SNMP アダプタを使用不可にする方法」
WBEM 用 SNMP アダプタにマッピングファイルディレクトリを再び強制的に読み取らせる	snmpXwbem.reg を更新して新しいサブツリーを組み込み pkill コマンドを使用することにより、WBEM 用 Solaris アダプタにマッピングファイルディレクトリを再び強制的に読み取らせる	45 ページの「WBEM 用 SNMP アダプタにマッピングファイルディレクトリを再び強制的に読み取らせる方法」

WBEM 用 SNMP アダプタのインストールと使用

この節では、WBEM 用 SNMP アダプタをインストール、起動、停止、および使用する方法について説明します。

▼ WBEM 用 SNMP アダプタをインストールする方法

- **Solaris** オペレーティング環境をシステムにインストールします。

アダプタソフトウェアは、Solaris ソフトウェアといっしょにシステムにインストールされます。

▼ WBEM 用 SNMP アダプタを起動する方法

SNMP アプリケーションを使用して CIM Object Manager からデータを取得する準備ができたなら、このあとの手順に従って、アダプタを起動します。

1. スーパーユーザーになります。
2. マスターエージェントを停止します。

```
# /etc/init.d/init.snmpdx stop
```
3. ディレクトリを /etc/snmp/conf に変更します。
4. snmpXwbem.rsrc- をアダプタリソースファイルに移動します。

```
# mv snmpXwbem.rsrc- snmpXwbem.rsrc
```
5. マスターエージェントを再起動します。

```
# /etc/init.d/init.snmpdx start
```

▼ WBEM 用 SNMP アダプタを使用不可にする方法

CIM Object Manager からのデータの取得が完了したとき、または /etc/snmp/conf にあるファイルを変更する場合には、アダプタを使用不可にする必要があります。次の手順に従ってください。

1. スーパーユーザーになります。
2. マスターエージェントを停止します。

```
# /etc/init.d/init.snmpdx stop
```
3. アダプタを停止します。

```
# /usr/bin/pkill -9 -x -u 0 snmpXwbemd
```
4. ディレクトリを /etc/snmp/conf に変更します。
5. snmpXwbem.rsrc の名前を一時的に変更します。

```
# mv snmpXwbem.rsrc snmpXwbem.rsrc-
```
6. マスターエージェントを再起動します。

```
# /etc/init.d/init.snmpdx start
```

▼ WBEM 用 SNMP アダプタにマッピングファイルディレクトリを再び強制的に読み取らせる方法

/var/sadm/wbem/snmp/map に新しいマッピングファイルを置いたり、そのディレクトリ内の既存のマッピングファイルを更新したりしたあとは、ディレクトリ内のすべてのマッピングファイルを再度読み取るようにアダプタに通知しなければなりません。アダプタに通知するには、シグナル SIGHUP を指定します。

次の手順に従って、アダプタにすべてのマッピングファイルを再び強制的に読み取らせてください (CIM Object Manager を再起動する必要はありません)。

1. スーパーユーザーになります。
2. 新しいマッピングファイル、またはマッピングファイル内の新しい項目が、アダプタによって登録されていないサブツリーを参照しているかどうか確認します。
 - 参照している場合は、次の手順に進みます。
 - 参照していない場合は、手順 5 に進みます。
3. /etc/snmp/conf/snmpXwbem.reg を更新して、新しいサブツリーを組み込みます。
4. マスターエージェントを停止し、再起動します。

```
# /etc/init.d/init.snmpdx stop
# /etc/init.d/init.snmpdx start
```
5. アダプタに、マッピングファイルを更新したことを通知します。

```
# /usr/bin/pkill -1 -x -u 0 snmpXwbemd
```

WBEM 用 SNMP アダプタに関する問題の障害追跡

この節では、アダプタを使用しているときに、発生する可能性のあるコンソールエラーメッセージを取り上げます。

この節で説明されていないエラー、問題、または予期しない結果が発生した場合、またはもっと綿密に障害追跡したい場合には、Solaris Management Console のログビューアを使ってログデータを調査してください。

Solaris Management Console のログビューアを起動する方法についての説明は、67 ページの「ログビューアを使ってログデータを調べる」を参照してください。

要求の送受信

エラーメッセージ

```
ERROR: sending request to Adapter Service.
```

```
ERROR: receiving request from Adapter Service.
```

原因

snmpXwbemd は WBEM が有効だと認識したにもかかわらず、Adapter Service と通信できなかったか、要求がタイムアウトになったかのいずれかです。

▼ 解決方法

別の要求を送信します。別の要求の送信にも失敗する場合は、要求および応答の FIFO に保留状態のメッセージが含まれていない (つまり、含まれているのが 0 バイトである) ことを確認します。

1. 次のコマンドを入力します。

```
# cd /var/sadm/wbem/snmp
```
2. 次のコマンドを入力します。

```
# ls -l
```

要求および応答の FIFO の一覧が表示されます。
3. 要求と応答のどちらかの **FIFO** に保留状態のメッセージが含まれて (**0** バイトより多く含まれて) いるかどうかを確認します。
含まれている場合、手順 4 に進みます。
含まれていない場合、次のようにします。
 - a. **Solaris Management Console** のログビューアを使ってログデータを調べ、問題を判別することにより、**WBEM** を停止させる必要があるかどうか確認します。
Solaris Management Console のログビューアを起動する方法についての説明は、67 ページの「ログビューアを使ってログデータを調べる」を参照してください。
 - b. 必要な場合は、**WBEM** を停止します。

```
# /etc/init.d/init.wbem stop
```
 - c. 手順 9 に進みます。
4. **WBEM** を停止します。

```
# /etc/init.d/init.wbem stop
```

5. マスターエージェントを停止します。
`# /etc/init.d/init.snmpdx stop`
6. FIFO のあるディレクトリに変更します。
`# cd /var/sadm/wbem/snmp`
7. 要求と応答の両方の FIFO を削除します。
`# rm _adapter_rcv.fifo`
`# rm _adapter_snd.fifo`
8. マスターエージェントを再起動します。
`# /etc/init.d/init.snmpdx start`
9. 手順 3 または 手順 4 で WBEM を停止した場合は、それを再起動します。
`# /etc/init.d/init.wbem start`

FIFO を開くことができない

エラーメッセージ

```
ERROR: request FIFO cannot be opened.
```

```
ERROR: response FIFO cannot be opened.
```

原因

アダプタが要求を受信したとき、またはアダプタが応答を処理したときに、プロトコルの問題が発生しました。

▼ 解決方法

別の要求を送信します。別の要求の送信にも失敗する場合は、要求および応答の FIFO に保留状態のメッセージが含まれていない (つまり、含まれているのが 0 バイトである) ことを確認します。

1. 次のコマンドを入力します。
`# cd /var/sadm/wbem/snmp`
2. 次のコマンドを入力します。
`# ls -l`
要求および応答の FIFO の一覧が表示されます。

3. 要求と応答のどちらかの **FIFO** に保留状態のメッセージが含まれて (**0** バイトより多く含まれて) いるかどうかを確認します。
含まれている場合、手順 4 に進みます。
含まれていない場合、次のようにします。
 - a. **Solaris Management Console** のログビューアを使ってログデータを調べ、問題を判別することにより、**WBEM** を停止させる必要があるかどうか確認します。
Solaris Management Console のログビューアを起動する方法についての説明は、67 ページの「ログビューアを使ってログデータを調べる」を参照してください。
 - b. 必要な場合は、**WBEM** を停止します。

```
# /etc/init.d/init.wbem stop
```
 - c. 手順 9 に進みます。
4. **WBEM** を停止します。

```
# /etc/init.d/init.wbem stop
```
5. マスターエージェントを停止します。

```
# /etc/init.d/init.snmpdx stop
```
6. **FIFO** のあるディレクトリに変更します。

```
# cd /var/sadm/wbem/snmp
```
7. 要求と応答の両方の **FIFO** を削除します。

```
# rm _adapter_rcv.fifo  
# rm _adapter_snd.fifo
```
8. マスターエージェントを再起動します。

```
# /etc/init.d/init.snmpdx start
```
9. 手順 3 または手順 4 で **WBEM** を停止した場合は、それを再起動します。

```
# /etc/init.d/init.wbem start
```

FIFO を作成できない

エラーメッセージ

```
ERROR: FIFO cannot be created.
```


原因

アダプタが要求または応答 FIFO を作成しようとした時にシステムエラーが発生しました。

解決方法

`/var/sadm/wbem/snmp` が存在すること、および書き込み権を持っていることを確認します。

WBEM サービスが起動されていない

エラーメッセージ

```
ERROR: WBEM Services are not started.
```

原因

マスターエージェントが、WBEM サービスが起動して実行中であるかどうかを検出できません。

解決方法

WBEM を再起動して、`snmpXwbem.reg` で `watch-dog-time` に設定された秒数だけ待ちます。

```
# /etc/init.d/init.wbem start
```

設定された秒数が経過すると、マスターエージェントがアダプタを自動的に起動します。デフォルトでは、マスターエージェントは 4 分 (または、`watch-dog-time` に設定された秒数) ごとにアダプタを起動しようとします。

注 - マスターエージェントがアダプタを自動的に起動するまで待てない場合は、マスターエージェントを停止してから再起動します。

```
# /etc/init.d/init.snmpdx stop
```

```
# /etc/init.d/init.snmpdx start
```

マスターエージェントは、ただちにアダプタを起動します。

SNMP プロバイダ

SNMP プロバイダは、管理要素についての情報を CIM Object Manager に提供するソフトウェアコンポーネントであり、提供する情報には、SNMP デバイスについての構成情報が含まれます。

注 - SNMP プロバイダは、ポート 162 で SNMP V1 および SNMP V2 トラップを待機することにより、トラップをサポートします。これらのトラップによって生成されたイベントを表示するには、`CIM_SNMPTrapIndication` を選択しなければなりません。

SNMP MIB ファイルから MOF ファイルを生成する

SNMP プロバイダを使用して Simple Network Management Protocol (SNMP) 情報にアクセスしたいときは、Management Information Base (MIB) ファイルを使用して Managed Object Format (MOF) ファイルを生成します。mib2mof コマンドは、CIM クラスに対して実行される MOF ファイル内の CIM 操作を、SNMP Provider が SNMP 操作に対応付けることを可能にする修飾子を生成します。mib2mof コマンドについては、mib2mof (1M) マニュアルページを参照してください。

注 - SNMP プロバイダは、SNMP トラップをサポートします。トラップは、CIM プロセスのインジケーションイベント `CIM_SNMPTrapIndication` として報告されます。クライアントがこのイベントのプロバイダに参加すると、そのプロバイダはポート 162 で SNMP V1 および SNMP V2 トラップを待機します。情報は、トラップからインジケーションにコピーされます。そのあと、インジケーションがクライアントに送信されます。

CIM スキーマおよび Solaris スキーマを記述した MOF ファイルは、`/usr/sadm/mof` にあります。MOF ファイルについては、『Solaris WBEM SDK 開発ガイド』を参照してください。

▼ SNMP MIB ファイルから MOF ファイルを生成する方法

1. スーパーユーザーになります。
2. 次のコマンドを入力します。

```
# mib2mof SNMP_MIB_filename  
次に例を示します。  
# /usr/sadm/bin/mib2mof sysctl1.mib
```


第 4 章

セキュリティの管理 (手順)

この章では、WBEM のセキュリティ機構と、CIM Object Manager の機能について説明します。

この章の内容は以下のとおりです。

- 59 ページの「Sun WBEM User Manager の使用 (作業マップ)」
- 53 ページの「WBEM のセキュリティ機構」
- 58 ページの「Sun WBEM User Manager を使ってアクセス制御を設定する」
- 63 ページの「WBEM のセキュリティに関する問題の障害追跡」

WBEM のセキュリティ機構

WBEM は、そのデータへのセキュリティ保護されたアクセスを確保するために、次のようないくつかの機構を採用しています。

- 認証 - クライアントのユーザー ID を WBEM サーバーに対して指定し、続いて、そのユーザーの資格を指定することによって、そのクライアントがその特定のユーザーであることを示すプロセスです。
- 役割の引き受け - WBEM サーバーが認証を検査する際に Solaris の役割によるアクセス制御 (RBAC) の役割 ID が WBEM サーバーによって使用されることを認めるプロセスです。
- セキュリティ保護されたメッセージング - それぞれのクライアント要求メッセージにセキュリティ保護されたメッセージオーセンティケータを追加するプロセスです。このオーセンティケータにより、WBEM サーバーはメッセージの送信元を検査し、メッセージが WBEM サーバーに届くまでの間にメッセージが変更されなかったかどうかを判別できます。
- 承認 - 認証されたユーザーまたは役割 ID が、それぞれの WBEM メソッド呼び出しによって管理されている WBEM データへのアクセス権を許可されていることを確認するプロセスです。承認の管理には、Solaris Management Console のユー

ツールや、Sun WBEM User Manager を使用します。

- 監査 - WBEM サーバーによって実行された特定の操作の監査レコードを書き込むプロセスです。これらのレコードは、認証されたユーザーが WBEM サーバーシステム上の管理データに加えた変更を追跡します。
- ロギング - 特定のセキュリティ関連イベントを WBEM ログに書き込むことで、WBEM ログは、Solaris Management Console のログビューアを使って参照することができます。

ここに挙げたそれぞれの機構について、このあと詳しく説明します。

認証

クライアントアプリケーションが CIM Object Manager サーバーに接続するときには、クライアントのユーザー ID が、WBEM サーバー上の CIM Object Manager によって認証されなければなりません。ユーザーの WBEM クライアントは、Solaris ユーザー ID と、それに伴うログインパスワードを提供する必要があります。その ID および資格は、クライアントと WBEM サーバーの間のセキュリティ認証交換で使用されて、クライアントが、WBEM サーバーシステムへのログインを許可された有効な Solaris ユーザーであることが確認されます。

WBEM サーバーがユーザーの ID および資格を検証できない場合、ユーザーの ID が無効である場合、WBEM サーバーは NO_SUCH_PRINCIPAL エラーの入った CIM セキュリティ例外を返します。

WBEM サーバーがユーザーの ID および資格を検証できない場合、ユーザーのパスワードがそのユーザーの ID に対して無効である場合、WBEM サーバーは INVALID_CREDENTIAL エラーの入った CIM セキュリティ例外を返します。

役割の引き受け

役割 ID を引き受けられるのは、WBEM ユーザーが Remote Method Invocation (RMI) プロトコルを選択した場合だけです。役割の引き受けは、XML/HTTP プロトコルではサポートされません。

WBEM の Solaris 実装は、クライアントが WBEM サーバー上の CIM Object Manager によって認証された場合に、クライアントが Solaris の役割の ID を引き受ける機能をサポートします。WBEM サーバーが RBAC 承認を使用してアクセス権の承認を検査する時、WBEM サーバーは、ユーザー ID 自身に許可されたアクセス権ではなく、引き受けた役割に許可されたアクセス権を検査します。

RBAC の役割については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

クライアントは、接続しようとするときに、Solaris のユーザー ID とパスワードに加えて、Solaris の役割 ID とパスワードを入力する必要があります。

WBEM サーバーが Solaris の役割 ID を検証できない場合、WBEM サーバーは、NO_SUCH_ROLE エラーの入った CIM セキュリティ例外を返します。

引き受けた役割 ID に対して役割のパスワードが無効な場合、WBEM サーバーは、INVALID_CREDENTIAL エラーの入った CIM セキュリティ例外を返します。

役割 ID とパスワードがどちらも有効な場合でも、そのユーザーが役割を引き受けることを許可されていない場合は、WBEM サーバーは、CANNOT_ASSUME_ROLE エラーの入った CIM セキュリティ例外を返します。

セキュリティ保護されたメッセージング

CIM RMI プロトコルでは、クライアントから WBEM サーバーへの各要求に、メッセージ内のデータパラメータから構築されたメッセージオーセンティケータが含まれています。さらに、一方向ダイジェストも作成され、認証交換の間に確立されたセッションキーもいっしょに作成されます。

WBEM サーバーは、このメッセージオーセンティケータを検証します。オーセンティケータは、認証された同じクライアントから要求が来たことを保証し、メッセージがサーバーまで通信される間に変更されたり再送信されたりしていないことを保証するものです。

メッセージが、変更または再送信されたものである場合、あるいは送信元のクライアントではないところで作成されたものである場合は、WBEM サーバーは、CHECKSUM_ERROR エラーの入った CIM セキュリティ例外を返します。さらに、WBEM サーバーは、ログメッセージを WBEM ログに書き込みます。

承認

WBEM サーバーが接続したあとは、WBEM サーバーは CIM クライアントとのそれ以降の操作におけるすべての承認検査について、認証されたユーザーまたは役割 ID を使用します。

WBEM は、次のものに基づく 2 種類の承認検査をサポートします。

- アクセス制御リスト (ACL)。これは、特定のネームスペースについて WBEM サーバーによって維持される
- RBAC 承認。これは、Solaris オペレーティング環境の一部として構成される

WBEM がどの承認検査機構を使用するかは、MOF クラスプロバイダがどのように実装されているかに応じて決まります。特定の MOF クラス操作について WBEM がどの承認検査機構を使用するかは、次のものに応じて決まります。

- WBEM が実行する操作
- MOF クラスの実装方法

Solaris Acl1.0.mof で定義されているクラスは、WBEM の ACL ベースのセキュリティを実装します。WBEM の ACL ベースのセキュリティは、Solaris WBEM Services に対してデフォルト承認スキーマを提供し、また、特定の状況下では、CIM 操作の特定のセットに適用されます。ACL ベースのセキュリティは、Solaris WBEM Services によって提供される固有の機能です。

WBEM サーバー上の特定のネームスペースについて ACL を確立するには、Sun WBEM User Manager (wbemadmin) を使用します。Sun WBEM User Manager では、ネームスペースについての ACL にユーザー名または役割名を追加したり、各ユーザーに「読み取り」または「書き込み」アクセス権を割り当てたりできます。Sun WBEM User Manager については、58 ページの「Sun WBEM User Manager を使ってアクセス制御を設定する」 および wbemadmin (1M) のマニュアルページを参照してください。

書き込みアクセス権のあるユーザーは、クラスのメタデータの変更、そのネームスペースにある MOF クラスのインスタンスの変更、およびインスタンスに対する呼び出しメソッドの発行が可能です。ローカル WBEM サーバーの root ユーザー ID には、いつでも、サーバー上のネームスペースすべてに対する書き込みアクセス権が許可されます。明示的な ACL 項目のない、すべての認証されたユーザーには、デフォルトで読み取りアクセス権が許可されます。

MOF クラスのメタデータのアクセスを含む操作 (たとえば、getClass) は、WBEM の ACL を使用します。これらの操作には、認証されたユーザーに、MOF クラスを含むネームスペースについての ACL によって許可されたアクセス権の検査が含まれます。ACL 項目に RBAC の役割を設定することもできますが、ACL 項目はいつもユーザー ID に対して検査され、役割 ID に対しては検査されません。つまり、ACL に役割名を設定することはできますが、CIM Object Manager は実行時にその役割名を検査しないということです。

MOF クラスのインスタンスが関係する操作には、WBEM ACL か RBAC 承認のいずれかの検査が含まれる可能性があります。

また、ユーザー、または役割 ID にアクセス権を許可して、プロバイダが RBAC 承認を使用している MOF クラスのインスタンスをそのユーザーがアクセスおよび変更できるようにすることもできます。これらのアクセス権を許可するには、Solaris Management Console のユーザーツールに含まれる権利ツールを使用します。ユーザーへのアクセス権の許可については、『Solaris のシステム管理 (セキュリティサービス)』の「権利プロファイルの作成または変更」を参照してください。

MOF クラスのインスタンスが WBEM の永続的なデータストアに格納されている場合、CIM Object Manager は、その MOF クラスを含むネームスペースについての WBEM ACL を検査します。MOF クラスプロバイダの実装がプロバイダのデータストアにアクセスする場合、または Solaris オペレーティング環境のシステムデータにアクセスする場合は、MOF クラスプロバイダの実装は、ほぼ必ず RBAC 承認検査を使用します。

一般に、MOF クラス定義にプロバイダ修飾子が含まれている場合、プロバイダの実装は通常、RBAC 承認検査を行います。MOF クラス定義にプロバイダ修飾子が含まれていない場合、CIM Object Manager は、次のことを行います。

- MOF クラスのインスタンスを WBEM の永続的なデータストアに格納する
- MOF クラスのネームスペースへのアクセスを制御する ACL を検査して、アクセスが許可されていることを確認する

監査

WBEM サーバーは、処理中に特定のイベントについて監査レコードを書き込みます。たとえば、クライアントの認証が成功または失敗したとき、およびユーザー情報を変更する操作が実行されたときにはいつでも、WBEM サーバーは監査レコードを書き込みます。

WBEM サーバーは、基本となる Solaris Basic Security Module (BSM) を使用して監査レコードを書き込みます。監査情報が確実に記録されるようにするには、WBEM サーバー上の Solaris オペレーティング環境で BSM の監査機構 (bsmconv) を有効にする必要があります。bsmconv コマンドについては、bsmconv (1M) のマニュアルページを参照してください。

注 – Trusted Solaris™ を使用している場合は、BSM の監査機構を有効にする必要はありません。

ロギング

WBEM サーバーは、特定のセキュリティイベントについて WBEM ログにログレコードを書き込みます。たとえば、クライアントに対する認証されたセッションが確立されたときや、承認検査が失敗したときなどです。WBEM ログは、Solaris Management Console のログビューアで調べることができます。ログビューアについては、第 5 章を参照してください。

セキュリティ関連のログイベントは、カテゴリが「セキュリティ (Security)」のログによって見分けることができます。カテゴリは、「カテゴリ (Category)」列に表示されます。セキュリティ関連のログメッセージだけを表示するには、ログビューアのログフィルタダイアログボックスで「セキュリティ (Security)」というカテゴリを選択します。ほとんどのセキュリティログメッセージには、クライアントのユーザー ID とクライアントホストの名前が含まれています。

Sun WBEM User Manager を使ってアクセス制御を設定する

Sun WBEM User Manager (wbemadmin) では、ユーザー自身や、特権を持つその他のユーザーが、次の操作を実行できます。

- 承認されたユーザーの追加および削除
- 承認されたユーザーのためのアクセス特権の設定
- ユーザー認証の管理と、WBEM 対応のシステム上の CIM オブジェクトへのアクセス

注 - アクセス制御を指定するユーザーは、Solaris ユーザーアカウントを持っていないければなりません。

Sun WBEM User Manager で実行できることと実行できないこと

アクセス特権は、個々のネームスペース、またはユーザーとネームスペースの組み合わせに対して設定できます。ユーザーを追加し、ネームスペースを選択すると、そのユーザーには、選択したネームスペース内の CIM オブジェクトに対する読み取り権がデフォルトで許可されます。

注 - ユーザーとネームスペースのアクセス権を効率的に組み合わせるには、まずネームスペースへのアクセスを制限してから、個々のユーザーにそのネームスペースに対する読み取り権、読み取り/書き込み権、または書き込み権を許可するようにします。

個々の管理対象オブジェクトにアクセス権を設定することはできません。しかし、あるネームスペース内のすべての管理対象オブジェクトにアクセス権を設定したり、ユーザーごとにアクセス権を設定したりすることはできます。

root としてログインすると、CIM オブジェクトに対して次のような種類のアクセス権を設定できます。

- 読み取りのみ - CIM スキーマオブジェクトへの読み取りのみのアクセス権を許可する。このアクセス権を持つユーザーは、インスタンスとクラスを取得することはできるが、CIM オブジェクトの作成、削除、および変更はできない
- 読み取り/書き込み - すべての CIM クラス、インスタンス、および呼び出しメソッドに対する完全な読み取り、書き込み、および削除のアクセス権を許可する

- 書き込み – すべての CIM クラスおよびインスタンスに対する書き込みおよび削除のアクセス権を許可するが、読み取り権は許可しない
- アクセス権なし – CIM クラスおよびインスタンスに対するアクセス権を許可しない

Sun WBEM User Manager の使用 (作業マップ)

次の表は、Sun WBEM User Manager を起動および使用するための手順を示しています。

作業	説明	参照先
Sun WBEM User Manager を起動する	wbemadmin コマンドを使って Sun WBEM User Manager を起動する	60 ページの「Sun WBEM User Manager を起動する方法」
ユーザーにデフォルトのアクセス権を許可する	Sun WBEM User Manager のユーザーアクセスツールを使用して、ユーザーにデフォルトのアクセス権を許可する	61 ページの「ユーザーにデフォルトのアクセス権を許可する方法」
ユーザーのアクセス権を変更する	Sun WBEM User Manager の「読み込み (Read)」および「書き込み (Write)」チェックボックスを使用して、ユーザーのアクセス権を変更する	61 ページの「ユーザーのアクセス権を変更する方法」
ユーザーのアクセス権を削除する	Sun WBEM User Manager のユーザーアクセスツールを使用して、ユーザーのアクセス権を削除する	61 ページの「ユーザーのアクセス権を削除する方法」
ネームスペースのアクセス権を設定する	Sun WBEM User Manager のネームスペースへのアクセスツールを使用して、ネームスペースのアクセス権を設定する	62 ページの「ネームスペースのアクセス権を設定する方法」
ネームスペースのアクセス権を削除する	Sun WBEM User Manager のネームスペースへのアクセスツールを使用して、ネームスペースのアクセス権を削除する	62 ページの「ネームスペースのアクセス権を削除する方法」

Sun WBEM User Manager の使用方法

この節では、Sun WBEM User Manager を起動して使用方法を説明します。

▼ Sun WBEM User Manager を起動する方法

1. スーパーユーザーになります。
2. コマンドウィンドウで、次のように入力します。

```
# /usr/sadm/bin/wbemadmin
```

Sun WBEM User Manager が起動し、「ログイン (Login)」ダイアログボックスが表示されます。

注 - 「ログイン (Login)」ダイアログボックスのいずれかのフィールドをクリックすると、そのコンテキストヘルプ情報が「コンテキストヘルプ (Context Help)」パネルに表示されます。

3. 「ログイン (Login)」ダイアログボックスの各フィールドに情報を入力します。
 - a. 「ユーザー名 (User Name)」フィールドに、ユーザー名を入力します。

注 - ログインするには、root\security ネームスペースに対する読み取り権が必要です。デフォルトでは、Solaris ユーザーはゲスト特権を持っているため、デフォルトのネームスペースに対する読み取り権が許可されています。読み取り権を持つユーザーは、ユーザー特権を表示することはできますが、変更することはできません。

ユーザーにアクセス権を許可するためには、root としてログインするか、root\security ネームスペースに対する書き込み権を持つユーザーとしてログインする必要があります。

- b. 「パスワード (Password)」フィールドに、ユーザーアカウントのパスワードを入力します。
4. 「了解 (OK)」をクリックします。

「User Manager」ダイアログボックスが開きます。このダイアログボックスには、ユーザー名と、現在のホスト上のネームスペース内の WBEM オブジェクトに対するアクセス権の一覧が表示されます。

▼ ユーザーにデフォルトのアクセス権を許可する方法

1. **Sun WBEM User Manager** を起動します。
2. ダイアログボックスの「ユーザーアクセス (**Users Access**)」の部分で、「追加 (**Add**)」をクリックします。
使用できるネームスペースの一覧を表示したダイアログボックスが開きます。
3. 「ユーザー名 (**User Name**)」フィールドに、**Solaris** ユーザーアカウントの名前を入力します。
4. 表示されたネームスペースの中からネームスペースを **1** つ選択します。
5. 「了解 (**OK**)」をクリックします。
このユーザー名が「**User Manager**」ダイアログボックスに追加されます。
6. 変更を保存し、「**User Manager**」ダイアログボックスを閉じるには、「了解 (**OK**)」をクリックします。変更を保存し、ダイアログボックスを開いたままにするには、「適用 (**Apply**)」をクリックします。
指定したユーザーに、選択したネームスペース内の CIM オブジェクトに対する読み取りアクセス権が許可されます。

▼ ユーザーのアクセス権を変更する方法

1. **Sun WBEM User Manager** を起動します。
2. アクセス権を変更するユーザーを選択します。
3. ユーザーに読み取りのみのアクセス権を許可するには、「読み取り (**Read**)」チェックボックスをクリックします。ユーザーに書き込み権を許可するには、「書き込み (**Write**)」チェックボックスをクリックします。
4. 変更を保存し、「**User Manager**」ダイアログボックスを閉じるには、「了解 (**OK**)」をクリックします。変更を保存し、ダイアログボックスを開いたままにするには、「適用 (**Apply**)」をクリックします。

▼ ユーザーのアクセス権を削除する方法

1. **Sun WBEM User Manager** を起動します。
2. ダイアログボックスの「ユーザーアクセス (**Users Access**)」の部分で、アクセス権を削除するユーザー名を選択します。
3. ネームスペースに対するこのユーザーのアクセス権を削除するために、「削除 (**Delete**)」をクリックします。

確認を求めるダイアログボックスが開きます。このダイアログボックスでは、ユーザーのアクセス権を本当に削除してよいかどうか確認を求められます。

4. 削除を確定するには、「了解 (OK)」をクリックします。
5. 変更を保存し、「User Manager」ダイアログボックスを閉じるには、「了解 (OK)」をクリックします。変更を保存し、ダイアログボックスを開いたままにするには、「適用 (Apply)」をクリックします。

▼ ネームスペースのアクセス権を設定する方法

1. Sun WBEM User Manager を起動します。
2. ダイアログボックスの「ネームスペースへのアクセス (Namespace Access)」の部分で、「追加 (Add)」をクリックします。
ダイアログボックスが開きます。このダイアログボックスには、使用可能なネームスペースの一覧が表示されます。
3. アクセス権を設定するネームスペースを選択します。

注 - デフォルトでは、ネームスペースに対する読み取り権のみがユーザーに許可されます。

- ネームスペースに対してアクセス権をまったく許可しない場合は、「読み取り (Read)」および「書き込み (Write)」の各チェックボックスをどちらも選択しません。
 - 書き込み権を許可する場合は、「書き込み (Write)」チェックボックスをクリックします。
 - 読み取り権を許可する場合は、「読み取り (Read)」チェックボックスをクリックします。
4. 変更を保存し、「User Manager」ダイアログボックスを閉じるには、「了解 (OK)」をクリックします。変更を保存し、ダイアログボックスを開いたままにするには、「適用 (Apply)」をクリックします。

▼ ネームスペースのアクセス権を削除する方法

1. Sun WBEM User Manager を起動します。
2. ダイアログボックスの「ネームスペースへのアクセス (Namespace Access)」の部分で、アクセス制御を削除するネームスペースを選択して、「削除 (Delete)」をクリックします。
ネームスペースからアクセス制御が削除され、「Sun WBEM User Manager」ダイアログボックスのネームスペースの一覧からそのネームスペースが削除されます。

3. 変更を保存し、「User Manager」ダイアログボックスを閉じるには、「了解 (OK)」をクリックします。変更を保存し、ダイアログボックスを開いたままにするには、「適用 (Apply)」をクリックします。

WBEM のセキュリティに関する問題の障害追跡

この節では、次のような場合の対応方法を説明します。

- クライアント (ユーザー) が WBEM サーバー上の CIM Object Manager によって認証されない
- WBEM サーバーが役割を引き受けない
- ACCESS_DENIED エラーが発生する

クライアント (ユーザー) が WBEM サーバー上の CIM Object Manager によって認証されない場合

WBEM サーバー上の CIM Object Manager によってクライアントが正しく認証されない場合、WBEM サーバーは、クライアントアプリケーションで CIM クライアントハンドルを確立しようとした時点で、CIM セキュリティ例外を返します。その例外には、認証の試行が失敗した理由を示すエラーコードが含まれています。

エラー	考えられる原因	解決方法
NO_SUCH_PRINCIPAL	指定されたユーザー ID が WBEM サーバー上の Solaris オペレーティング環境で無効であるか、そのユーザーのユーザーアカウントがパスワードを持っていない、またはロックされている	ユーザーが有効なユーザー ID を持っていることを確認する。つまり、ユーザーが、WBEM サーバーマシン上の Solaris オペレーティング環境にログインできることを確かめる。WBEM サーバーとして設定された Solaris システムはサーバー上に構成されたネームサービスから取得したユーザー ID を使用していることが考えられるので、ネームサービスのテーブルを確認する必要があるかもしれない

エラー	考えられる原因	解決方法
INVALID_CREDENTIAL	指定されたユーザー (または、役割 ID を引き受けている場合は、役割) のパスワードが、WBEM サーバー上の Solaris オペレーティング環境内の該当ユーザーに対して無効である	ユーザーのパスワードが正しいかどうかを確認する
NO_SUCH_ROLE	WBEM サーバーに対する認証で引き受けられた役割 ID が、WBEM サーバー上の Solaris オペレーティング環境で有効な RBAC 役割ではない	<p>役割 ID は、サーバー上の passwd テーブル内の有効な項目かもしれないが、その ID ではサーバーにログインできない (Solaris では、役割 ID で直接ログインすることはできない)。したがって、passwd テーブルで役割 ID を確認し、user_attr テーブルを確認してその役割がユーザーの役割の種類として定義されていることを確認する。user_attr テーブル内の役割 ID には、それぞれ、type=role という構文の属性が含まれている</p> <p>あるいは、Solaris Management Console のユーザーツールを使って、有効なユーザーまたは有効な役割 ID を確認することもできる。ユーザーアカウントツールではユーザーの確認、管理役割ツールでは役割の確認ができる。ただし、ユーザーツールを使用するときには、CIM Object Manager サーバー上のテーブルの正しいソースを知っていなければならない。つまり、CIM Object Manager サーバーが NIS などのネームサービスを使用している場合は、そのネームサービスのマスターサーバーにアクセスしなければならない</p>

エラー	考えられる原因	解決方法
CANNOT_ASSUME_ROLE	役割 ID は有効だが、認証交換で指定されたユーザー ID は、その役割を引き受けるように構成されていない	Solaris Management Console ユーザーツールコレクションに含まれている管理役割ツールを使って、ユーザーに役割を明示的に割り当てる。ツールについては、『Solaris のシステム管理 (セキュリティサービス)』の「役割プロパティの変更」を参照

その他の CIM セキュリティ例外エラーが起きる場合

WBEM サーバーは、その他のエラーインジケーションを CIM セキュリティ例外に入れて返すことがあります。しかし、これらのインジケーションの多くは、認証交換におけるシステム障害を示すものです。認証交換のセキュリティオプションについて、WBEM クライアントの構成が WBEM サーバーの構成と互換性がない可能性が考えられます。

これらのエラーインジケーションが発生した場合は、クライアントマシン上の WBEM インストールで、`WbemClient.properties` にセキュリティの適切な構成プロパティ値が入っているかどうか確認します。このファイルは通常、WBEM のインストールディレクトリ内のベンダー拡張サブディレクトリ `/usr/sadm/lib/wbem/extension` にあります。

さらに、クライアントアプリケーションの `CLASSPATH` の設定を確認して、`sunwbem.jar` と拡張ディレクトリのパス名がクラスパス上にあることを確認します。

認証検査に失敗する

WBEM サーバーに対する要求に関連したデータをアクセスまたは変更することについてクライアントが承認されていない場合、WBEM サーバーは、`ACCESS_DENIED` エラーの含まれた CIM セキュリティ例外をその要求に対して返します。

`ACCESS_DENIED` エラーは、認証されたユーザーまたは役割が、その要求によって管理されているデータに対する適切なアクセス権を許可されていないため、WBEM 要求が完了できなかったことを示します。

WBEM ログのセキュリティメッセージから、失敗した要求のメッセージを見つけます (ログデータを調べる方法については、67 ページの「ログビューアを使ってログデータを調べる」を参照してください)。承認障害のログメッセージは、「概要 (Summary)」列に「アクセスを拒否しました (Access denied)」と示されています。

す。「ユーザー (User)」列には、検査に使用された認証されたユーザーまたは役割の名前が表示されます。「ソース (Source)」列には、検査を実行しているプロバイダの名前が表示されます。この列に表示されるプロバイダ名は、プロバイダ実装のクラス名ではなく、ユーザーにわかりやすい名前です。

詳細なメッセージの中に、検査されているアクセス権の名前や、ユーザーまたは役割に許可されていなかったアクセス権の名前が含まれています。

アクセス権が「*namespace:right*」という形式で表示されている場合、承認検査ではネームスペース ACL が使用されたことを示します。認証されたユーザーは、そのネームスペースに対してそのアクセス権 (読み取りまたは書き込み) を許可されていません。

Sun WBEM User Manager (wbemadmin) を使用して、ユーザーに適切なアクセス権を許可してください。Sun WBEM User Manager については、58 ページの「Sun WBEM User Manager を使ってアクセス制御を設定する」を参照してください。

アクセス権が「*solaris.application.right*」という形式で表示されている場合、承認検査では RBAC 承認が使用されました。

Solaris Management Console ユーザーツールコレクションに含まれている管理役割ツールを使って、ユーザーまたは役割に必要なアクセス権を許可してください。この手順については、『Solaris のシステム管理 (セキュリティサービス)』の「役割プロパティの変更」を参照してください。

第 5 章

システムログデータの表示 (手順)

WBEM ログファイルを使用すると、システム管理者は、管理サブシステムが生成したエラー、警告、および情報メッセージを追跡できます。また、WBEM ログインサービスは、アプリケーション開発者やプロバイダの作成者がログファイルにログメッセージを書き込むことを可能にします。たとえば、システムがシリアルポートにアクセスできないとき、システムが正常にファイルシステムをマウントできたとき、またはシステムで実行中のプロセスの数が許容数を超えたときに、ログメッセージを書き込むことができます。

この章では、ログデータを表示する方法を説明します。

ログビューアを使ってログデータを調べる

ログレコードを作成したあとは、Solaris Management Console アプリケーションとログビューアを起動することができます。ログレコードは、Solaris Management Console ソフトウェアを起動した時に自動的に作成されます。

ログレコードの詳細のすべてを Solaris Management Console のログビューアで見ることができます。

▼ Solaris Management Console アプリケーションとログビューアを起動する方法

1. 次のコマンドを入力します。

```
$ smc
```

2. 「ナビゲーション (Navigation)」パネルで、「このコンピュータ (This Computer)」をダブルクリックするか、その横にある展開 / 圧縮のアイコンをシングルクリックします。
「このコンピュータ (This Computer)」の下に、コマンドのツリーが表示されます。
3. 「システムステータス (System Status)」をダブルクリックします。
「ログビューア (Log Viewer)」アイコンが表示されます。
4. 「ログビューア (Log Viewer)」アイコンをクリックします。
ログビューアが起動します。

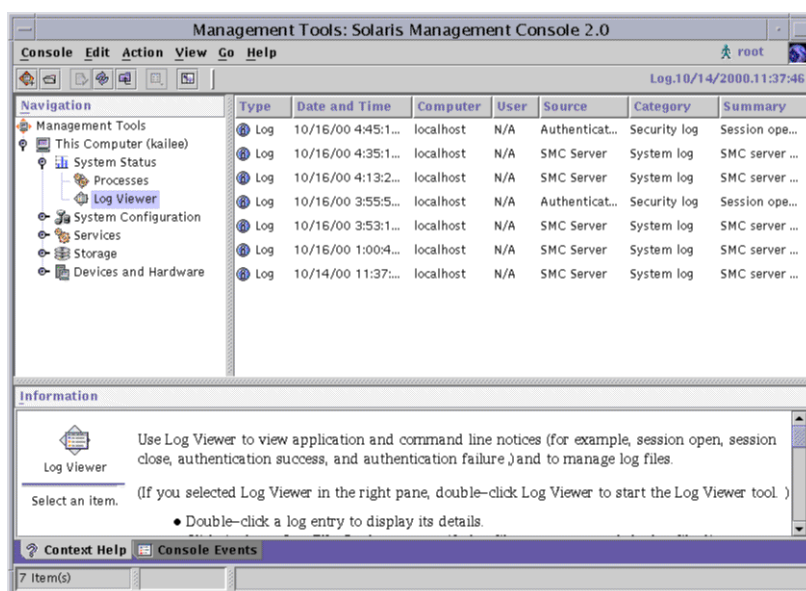


図 5-1 Solaris Management Console アプリケーションで「ログビューア (Log Viewer)」を選択した状態

索引

数字・記号

- 050SUNWwbccou.map, 38
 - 内容の構文, 40
 - ファイル名の構文, 40

C

- CIM (Common Information Model)
 - 概要, 16, 17
- CIM Object Manager
 - Repository, 21, 25, 26, 28
 - 起動時の機能, 25
 - 再起動, 27
 - 停止, 27
 - プロバイダの使用方法, 23
 - プロバイダを使用する方法, 23, 25
- CIM オブジェクト, 定義, 16
- CIM スキーマ, 17, 50
 - 共通モデル, 17
 - コアモデル, 17
- Common Information Model (CIM)
 - 概要, 16, 17

D

- Distributed Management Task Force (DMTF), 16

I

- init.wbem コマンド, 26

J

- Java, Managed Object Format (MOF) からの変換, 21
- Java Native Interface, 22
- JavaSpaces データストア, 保存, 29

M

- Managed Object Format (MOF), Java への変換, 21
 - MOF ファイルも参照
- mib2mof コマンド、構文, 50
- MIB ファイル、MOF ファイルを生成する方法, 50
- MOF ファイル
 - 生成, 50

R

- Repository
 - CIM Object Manager, 21, 25, 26, 28

S

- snmpXwbem.acl, 36

- snmpXwbem.reg, 36
- snmpXwbem.rsrc, 38
- SNMP プロバイダ
 - CIM_SNMPTrapIndication, 50
 - MOF ファイル, 50
 - 説明, 50
 - トラップ, 50
 - ポート 162, 50
- Software Developer's Kit (SDK), 24
- Solaris WBEM Services, 18
- Solaris スキーマ, 17, 22, 50
- Solaris プロバイダ, 22
- Solstice Enterprise Agents, マスターエージェント, 33
- Sun SNMP Agent, WBEM 用 SNMP アダプタとの比較, 35
- Sun WBEM User Manager
 - 起動, 60
 - デフォルトのアクセス権, 61
 - ネームスペースのアクセス権を削除する, 62
 - ネームスペースのアクセス権を設定する, 62
 - ユーザー特権の設定, 58
 - ユーザーのアクセス権の変更, 61
 - ユーザーのアクセス権を削除する, 61

W

- WBEM (Web-Based Enterprise Management)
 - 互換性, 15
 - サポートされる標準, 15
 - 定義, 15
- WBEM Services, Solaris, 18
- WBEM セキュリティ, エラーメッセージ, 63
- WBEM 用 SNMP アダプタ
 - SNMP から CIM オブジェクトへのマッピング, 35
 - Sun SNMP Agent との比較, 35
 - インストール, 43
 - エラーメッセージ
 - FIFO を作成できない, 48
 - FIFO を開くことができない, 47
 - WBEM サービスが起動されない, 49
 - 要求の送受信, 46
 - 起動, 44
 - 構成, 35

WBEM 用 SNMP アダプタ (続き)

- 使用不可, 44
- 説明, 33
- 停止, 44
- 内部動作の説明, 34
- マッピングファイル
 - Solaris の内容, 38
 - 内容の構文, 40
 - ファイル名の構文, 40
- マッピングファイルディレクトリの再読み取り, 45

X

- XML, ほかの WBEM システムとの相互運用性, 24

あ

- アップグレード, CIM Object Manager Repository, 28
- アプリケーションプログラミングインタフェース (API), プロバイダ, 23

え

- エラーメッセージ
 - WBEM セキュリティ, 63
 - WBEM 用 SNMP アダプタ
 - FIFO を作成できない, 48
 - FIFO を開くことができない, 47
 - WBEM サービスが起動されない, 49
 - 要求の送受信, 46

き

- 起動時の機能, 25
- 共通モデル, 基底クラス, 17

こ

- コアモデル, 17

コマンド

init.wbem, 26
mib2mof, 50
wbemadmin, 60

し

障害追跡

エラーメッセージを参照

す

スキーマ

CIM, 17, 50
Solaris, 17, 22, 50
定義, 16

せ

セキュリティ, Sun WBEM User Manager, 58

そ

ソフトウェアコンポーネント, 19

と

動的データ, 23

特権

Sun WBEM User Manager, 58
デフォルトのアクセス権をユーザーに許可する, 61

ね

ネームスペース

定義, 22
デフォルト, 23

ひ

標準、WBEM によってサポートされる, 15

ふ

プロバイダ

CIM Object Manager の再起動, 27
SNMP, 50
Solaris, 22
機能, 23
定義, 22
ネイティブの記述, 22

ほ

ほかの WBEM システムとの相互運用性, XML, 24

ほかの標準との互換性, WBEM, 15

ま

マスターエージェント, 33

ろ

ログデータ, 表示, 67

