Solaris Container Manager 3.6 インストールと管理

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

Part No: 819–4977–10
2005 年 12 月

Copyright 2005 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

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

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

U.S. Government Rights Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

本製品に含まれる HG-MinchoL、HG-MinchoL-Sun、HG-PMinchoL-Sun、HG-GothicB、HG-GothicB-Sun、および HG-PGothicB-Sun は、株式会社リコーがリョービイマジクス株式会社からライセンス供与されたタイプフェースマスタをもとに作成されたものです。 HeiseiMin-W3H 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ー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、JDK、および N1 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、登録商標もしくは、サービスマークです。

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。

Netscape™、Netscape Navigator™、および Mozilla™ は、米国およびその他の国における Netscape Communications Corporation の商標および登録商標です。

ORACLE® は、Oracle Corporation の登録商標です。

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。

「ATOK Server/ATOK12」に含まれる郵便番号辞書 (7 桁/5 桁) は日本郵政公社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)。

「ATOK Server/ATOK12」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』に添付のものを使用しています。

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、OPEN LOOK のグラフィカル・ユーザインタフェースを実装するか、またはその他の方法で米国 Sun Microsystems 社との書面によるライセンス契約を遵守する、米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

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

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: Installing and Administering Solaris Container Manager 3.6

Part No: 819-4099–10

051206@13215

目次

はじめに　**13**

1　Solaris Container Manager 3.6 の紹介　**17**
Container Manager の概要　18
Container Manager およびその他のリソース管理ユーティリティ　18
Solaris Container モデル　19
リソース管理　21
リソースプール　22
リソースの制御　23
ゾーン　23
大域ゾーン　23
非大域ゾーン　24
リソース使用状況レポートおよび拡張アカウンティングデータ　24
Container Manager をご使用になる前に　25
サーバー統合　25
Container Manager 使用例　26
Oracle 用のゾーンがある複数のプロジェクト　26
動的リソースプールの例　26
同一コンテナを共有するアプリケーション　26
複数のシステム上の Oracle 10g ラック　26
複数のリソースプールがある複数のシステム　27
Solaris Container Manager 3.6 の新機能と変更点　27
ゾーン管理　28
動的リソースプール　28
IPQoS を使用した帯域幅制御　28
柔軟性のあるプロセス管理　28

タイムシェアスケジューラ　29
コンテナの強化機能　29
Container Manager のマニュアル　29
基本的な使用方法　30

2　Container Manager のインストールと設定　31
Container Manager ソフトウェアについて　31
Container Manager ソフトウェアのインストール　35
▼ Sun Management Center 3.6 へのアップグレード時に Container Manager をインストールする　35
▼ Container Manager を別個にインストールする　35
Container Manager ソフトウェアの設定　37
▼ Sun Management Center のインストールまたはアップグレード時に Container Manager を設定する　39
▼ Container Manager を別個に設定する　39
プロファイルの作成　41
バックアップと復元　41
エージェント更新　42
Container Manager ソフトウェアのアンインストール　42
▼ Container Manager ソフトウェアを削除する　42

3　コンテナの概要と製品の起動　45
コンテナ の概要　45
コンテナのプロパティー　47
プロジェクトの状態　49
コンテナとプロジェクト　49
プロジェクトの有効化　50
プロジェクトの無効化　50
Container Manager の GUI　50
▼ Container Manager の GUI を起動する　51
▼ Java Web Console を再起動する　52
Container Manager の GUI のタブ　52
ホスト表示　54
コンテナ表示　57
ホストとコンテナのグループ化　58
▼ コンテナグループまたはホストグループを作成する　58
▼ コンテナまたはホストを別のグループに移動する　58
デフォルトのコンテナ　59

コンテナの作成　61
プロジェクトの種類　61
リソース予約 (CPU シェア数)　62

4　プロジェクトの管理　**73**
プロジェクトの作成　73
▼ 新規プロジェクトウィザードを起動する　74
ユーザーベースまたはグループベースのプロジェクトの作成　77
▼ ユーザーベースまたはグループベースのプロジェクトを作成する　78
アプリケーションベースのプロジェクトの作成　80
▼ アプリケーションの一致式を決定する　81
▼ アプリケーションベースのプロジェクトを作成する　82
プロジェクト内でのプロセスの移動または起動　85
▼ プロセスをアプリケーションベースのプロジェクトに個別に移動する　86
▼ プロジェクト内でアプリケーションを起動する　87
プロジェクトの有効化と無効化　88
▼ プロジェクトを有効にする　89
▼ 無効なプロジェクトを有効にする　90
▼ 有効なプロジェクトを無効にする　90
プロジェクトのプロセスの表示　91
▼ プロジェクト内で実行中のプロセスを「ホスト」表示で表示する　92
▼ プロジェクト内で実行中のプロセスを「コンテナ」表示で表示する　93
コンテナおよびプロジェクトの変更　93
▼ プロパティシートを使用してコンテナを変更する　95
▼ プロパティシートを使用してプロジェクトを変更する　96
リソース変更ジョブによるプロジェクトの変更　97
▼ リソース変更ジョブを使用してプロジェクトを変更する　99
▼ 待ち状態のリソース変更ジョブを変更する　100
▼ リソース変更ジョブのログを表示する　101
プロジェクトの削除　101
▼ コンテナを削除する　102

5　リソースプールの管理　**103**
リソースプールの概要　103
新規リソースプールの作成　104
▼ 新しいリソースプールを作成する　104
リソースプールの変更　106

▼ リソースプールを変更する 107
リソースプールの削除 107
▼ リソースプールを削除する 107

6 ゾーンの管理 **109**
ゾーンの概要 109
非大域ゾーンの状態 110
非大域ゾーンの作成 110
始める前に 111
▼ 非大域ゾーンを作成する 111
非大域ゾーンのコピー 116
▼ 非大域ゾーンをコピーする 116
非大域ゾーンの削除、起動、停止 117
▼ 非大域ゾーンを削除、起動、または停止する 118
ゾーンのログファイルの表示 118
▼ ゾーンのログファイルを表示する 118

7 アラームの管理 **119**
アラームの管理 119
アラームの設定 119
▼ アラームのしきい値を設定する 120
▼ アラームのしきい値を削除する 121
▼ 未確認のアラームを表示する 121

8 レポートの作成と拡張アカウンティングデータの使用 **123**
レポート 123
データ収集プロセス 126
レポート要求 126
▼ ホストのリソース使用状況レポートを要求する 127
▼ 有効なプロジェクトのリソース使用状況レポートを要求する 128
▼ コンテナのリソース使用状況レポートを要求する 129
▼ リソースプールのリソース使用状況レポートを要求する 129
▼ (Solaris 10 のみ) ゾーンのリソース使用状況レポートを要求する 131

A コマンド行インストール **133**

Container Manager ソフトウェアのインストール 133

▼ Sun Management Center のインストール時に Container Manager をインストールする 134

▼ Sun Management Center 3.6 へのアップグレード時に Container Manager をインストールする 134

▼ Container Manager を別個にインストールする 135

Container Manager の設定 136

▼ Sun Management Center のインストールまたはアップグレード時に Container Manager を設定する 136

▼ Container Manager を別個に設定する 136

Container Manager ソフトウェアの削除 139

▼ `es-uninst` を使用して Container Manager を削除する 139

用語集 **141**

索引 **143**

表目次

表 1–1 Solaris Container Manager の新機能 27
表 1–2 関連マニュアル 29
表 2–1 Sun Management Center と Solaris Container Manager のシステム要件 32
表 2–2 Solaris OS のバージョン別の Container Manager の機能 34
表 3–1 Container Manager の GUI のタブ 53
表 3–2 プロジェクトの種類の詳細 62
表 4–1 プロパティシートの詳細 94
表 4–2 リソース変更ジョブのスケジュールの例 98
表 7–1 アラームのしきい値の名前 120
表 8–1 使用状況レポートの種類 124

図目次

図 1–1 Solaris Container の例 20
図 1–2 ホスト上のコンテナの例 20
図 1–3 ホスト上のプロジェクトの例 22
図 3–1 コンテナとプロジェクト 48
図 3–2 プロジェクトの状態 49
図 3–3 Java Web Console のログインページ 51
図 3–4 Container Manager のメインページ 52
図 3–5 例:「プロジェクト」表が表示された「ホスト」表示 55
図 3–6 例: デフォルトコンテナに関連付けられているホストが表示された「コンテナ」表示 57
図 3–7 例:「System」コンテナグループのコンテナの表示 60
図 3–8 プロジェクトの CPU シェア数 64
図 3–9 CPU シェア数 65
図 3–10 ゾーンのシェア数 66
図 3–11 ゾーンの CPU シェア数 67
図 3–12 プロジェクトの CPU シェア数 68
図 4–1 例: リソース予約とリソースプールを変更するプロパティシート 94
図 4–2 例: コンテナを変更するプロパティシート 95
図 4–3 例:「リソース変更ジョブ」表 100
図 5–1 Solaris 10 のホストの「リソースプール」ウィンドウ 105
図 6–1 ゾーン作成時のパラメータのウィンドウ 111
図 6–2 ゾーンの属性のウィンドウ 113
図 6–3 CPU シェア数のウィンドウ 114
図 6–4 IPQoS の属性のウィンドウ 115
図 6–5 ゾーンのコピーのウィンドウ 117

はじめに

『Solaris Container Manager 3.6 インストールと管理』では、Container Manager を使ってコンテナを作成、使用、および管理する方法について説明します。

注 – この製品のバージョン 1.0 は、N1™ Grid Console - Container Manager という名前でした。

注 – Solaris™ リリース 10 は、SPARC® および x86 ファミリのプロセッサアーキテクチャー (UltraSPARC®、SPARC64、AMD64、Pentium、Xeon EM64T) を使用するシステムをサポートします。サポートされているシステムについては、Solaris 10 Hardware Compatibility List (`http://www.sun.com/bigadmin/hcl`) を参照してください。本書では、異なるプラットフォーム間の実装の違いについても言及します。

本書では、「x86」という用語は AMD64 あるいは Intel Xeon/Pentium 製品系列と互換性のあるプロセッサを使用して製造された 32 ビットおよび 64 ビットシステムを意味します。サポートされるシステムについては、Solaris 10 Hardware Compatibility List を参照してください。

対象読者

本書は、Sun™ Management Center 製品に精通しているユーザーを対象とします。したがって、本書では、Sun Management Center に関連する多くの用語や概念について説明していません。Sun Management Center については、『Sun Management Center 3.6 インストールと構成ガイド』 および『Sun Management Center 3.6 ユーザーガイド』を参照してください。

内容の紹介

本書では、Container Manager ソフトウェアのインストールと使用方法について説明します。

第 1 章では、Container Manager の概要を示します。

第 2 章では、インストールと設定の方法を示します。

第 3 章では、コンテナモデルと製品の起動方法について説明します。

第 4 章では、プロジェクトを作成、使用、および管理する方法を示します。

第 5 章では、リソースプールを作成、使用、および管理する方法を示します。

第 6 章では、ゾーンを作成、使用、および管理する方法を示します。

第 7 章では、アラームの使用方法を示します。

第 8 章では、レポートの作成方法を示します。

付録 A では、コマンド行を使用してインストールと設定を行う方法を示します。

用語集では、本書で使用する用語と、その定義を説明します。

関連情報

次のマニュアルや書籍では、Container Manager ソフトウェアを使うときに役立つ概念について説明しています。

- 『Solaris Resource Manager 1.3 のシステム管理』
- 『Solaris のシステム管理 (ネットワークサービス)』
- 『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』
- David Hornby、Ken Pepple 共著『Consolidation in the Data Center』(Sun Blueprints)
- Sun ホワイトペーパー『Consolidating Oracle RDBMS Instances Using Solaris Resource Manager Software』

Sun Management Center ソフトウェアとアドオン製品の最新情報については、`http://www.sun.com/solaris/sunmanagementcenter` を参照してください。

Sun のオンラインマニュアル

docs.sun.com[SM] 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。

Sun のマニュアルの注文方法

Sun Microsystems は製品の印刷されたマニュアルを提供しています。それらのマニュアルの一覧と注文方法については、http://docs.sun.com の「印刷された文書の購入」を参照してください。

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。	`.login` ファイルを編集します。 `ls -a` を使用してすべてのファイルを表示します。 `machine_name% you have mail.`
`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。	`machine_name%` **`su`** `Password:`
AaBbCc123	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、 `rm` *filename* と入力します。

表 **P–1** 表記上の規則　　(続き)

字体または記号	意味	例
『 』	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。	`sun% grep '^#define \ XV_VERSION_STRING'`

コマンド例のシェルプロンプト

以下の表に、C シェル、Bourne シェル、および Korn シェルのデフォルトのシステムプロンプト、およびスーパーユーザーのプロンプトを紹介します。

表 **P–2** シェルプロンプト

シェル	プロンプト
C シェル	`machine_name%`
C シェルのスーパーユーザー	`machine_name#`
Bourne シェルおよび Korn シェル	`$`
Bourne シェルおよび Korn シェルのスーパーユーザー	`#`

第 1 章

Solaris Container Manager 3.6 の紹介

この章では、Solaris Container Manager 3.6 (Container Manager) の概要を示します。

この章では、以下の内容について説明します。

- 18 ページの「Container Manager の概要」
- 19 ページの「Solaris Container モデル」
- 21 ページの「リソース管理」
- 23 ページの「ゾーン」
- 24 ページの「リソース使用状況レポートおよび拡張アカウンティングデータ」
- 25 ページの「Container Manager をご使用になる前に」
- 26 ページの「Container Manager 使用例」
- 27 ページの「Solaris Container Manager 3.6 の新機能と変更点」
  - 28 ページの「ゾーン管理」
  - 28 ページの「動的リソースプール」
  - 28 ページの「IPQoS を使用した帯域幅制御」
  - 28 ページの「柔軟性のあるプロセス管理」
  - 29 ページの「タイムシェアスケジューラ」
  - 29 ページの「コンテナの強化機能」
- 29 ページの「Container Manager のマニュアル」
- 30 ページの「基本的な使用方法」

Container Manager の概要

Solaris Container Manager 3.6 は、Sun Management Center 3.6 リリースのアドオン製品です。この製品を使用すると、サーバーを統合し、多数のサーバーとソフトウェアを使用するネットワークのコストを抑えることができます。Container Manager では、コンテナ、プロジェクト、リソースプール、およびゾーンを作成および管理できます。ハードウェアリソースの使用率が上がり、またサーバー数に対する管理者数を減らすことができるという利点があります。

この製品では、次の作業を行うことができます。

- ホストのリソースのパーティション分割
- リソースの割り当て、制御、および整理
- アプリケーションの他のアプリケーションからの分離
- 特定のアプリケーションのリソースの消費状況の解析
- リソースの使用状況の監視、CPU およびメモリーの使用状況の拡張アカウンティングデータの収集

IPアドレス、ディスク記憶装置、およびアプリケーションを含めて、ユーザーが独自の仮想的環境を必要とするような組織にとって、コンテナは理想的です。たとえば、メールサーバー、Web サーバー、データベースなどの特定のアプリケーション用のコンテナを設定できます。また、米国、南北アメリカ、ヨーロッパ、アジア太平洋などの地域用のコンテナを設定することもできます。同様に、人事、研究開発、営業などの機能部門用のコンテナを設定することもできます。

業界によっても、コンテナまたはゾーンを使用する目的が異なります。大学は、学生ごとにゾーンを設定し、OS のインスタンス、システムリソースの共有、および root パスワードを割り当てることができます。ワイヤレス企業は、長距離通話サービス、市内通話サービス、ボイスメールなどのサービスを監視するコンテナを設定できます。ケーブル会社やインターネットサービスプロバイダは、DSL、ケーブルモデム、またはケーブルテレビサービス用のコンテナを設定できます。金融機関は、データウェアハウスに対して複雑な照会を実行するユーザーと、オンライントランザクション処理が必要なユーザーに別々のコンテナを設定できます。独立ソフトウェアベンダー (ISV) は、ソフトウェアやサービスを販売する顧客ごとにコンテナまたはゾーンを設定できます。

Container Manager およびその他のリソース管理ユーティリティ

この製品では、Solaris 8、Solaris 9、および Solaris 10 で動作する既存のリソース管理ユーティリティがまとめられます。具体的には、この製品には、Solaris Resource Manager 1.3 と Solaris 9 Resource Manager の設定を簡素化するツールが含まれます。

Solaris リソース管理ユーティリティに関する詳細は、『Solaris Resource Manager 1.3 のシステム管理』および『Solaris のシステム管理 (ネットワークサービス)』を参照してください。

Solaris Container モデル

Solaris Container は、物理的なシステムリソースの集合を体系づけて管理するのに役立つ抽象化層です。コンテナを使用して、アプリケーションに必要なリソース要件の詳細な計画を立てることができます。アプリケーションに対するリソース要件が、Solaris Container モデルの中心となります。このモデルでは、サービスまたは作業負荷に注目します。サービスは、アプリケーションによってもたらされ、システムへの作業負荷となります。実行中のアプリケーションなど、関連する一連のプロセスが作業負荷となります。

Solaris Resource Manager 1.3 では、作業負荷に基づいた古い方式の管理が実装されていました。このリリースでは、作業負荷が制限ノード (`lnode`) に関連付けられていました。Container Manager ソフトウェアでは、この方式を拡張しています。現在のコンテナモデルは、サービスに対してリソースの継続的な割り当てを構成および管理する手段を提供しています。一般的なサービスの例には、給与、顧客の注文の検索、Web サービスの配信があります。

サーバー統合では、アプリケーションを制限する環境を記述できる必要があります。この記述を設定すると、サーバー当たり 1 つのアプリケーションを実行する環境から、単一のサーバーで複数のアプリケーションを実行する環境に移行できます。コンテナは、この記述であると同時に、そのインスタンス化でもあります。たとえば、簡単なコンテナでは、CPU、物理メモリー、および帯域幅などのシステムリソースを記述できます。複雑なコンテナでは、セキュリティ、名前空間の独立、およびアプリケーション障害も制御できます。

次の Solaris Container の図に、サービスとリソースの関係を示します。

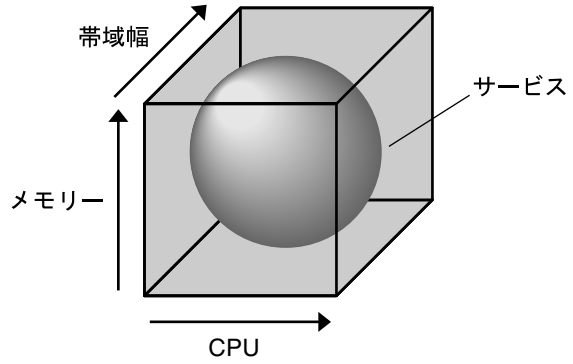


図 **1–1** Solaris Container の例

枠がコンテナを表します。サービスを囲む枠の x、y、および z の各軸に沿って 3 種類のリソースがあります。このモデルでは、CPU、メモリー、および帯域幅が基本リソースです。サービスは、コンテナにどのように包含されるかを示すために枠で囲まれています。このリリースでは、Container Manager によって 3 つの基本リソースである CPU、物理メモリー、および帯域幅がすべて制御されます。

Container Manager は作業負荷に注目するため、単一のホストで使用されるリソース量は監視されません。ホストとは、Container Manager のエージェントソフトウェアがインストールされ、また Sun Management Center サーバーのコンテキストに含まれるシステムです。インストールが完了すると、ホストは自動的に検出され、その名前が「ホスト」表示のナビゲーションウィンドウに追加されます。ソフトウェアによって、サービスで使用されるリソース量が監視されます。このモデルでは、サービスのインスタンス 1 つが、個別のホストで実行される 1 つ以上のプロセスを表します。データは、システムの健全性の監視とアカウンティングのために保持されます。

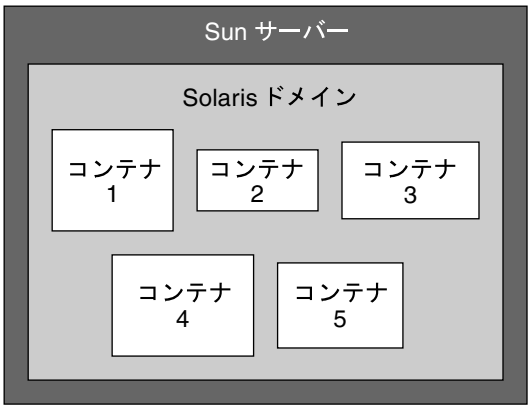


図 **1–2** ホスト上のコンテナの例

単一のホストで同時に複数のコンテナを有効にできます。単一のホストに複数のコンテナがある場合、ホストがコンテナを拡張および縮小できるように、コンテナの境界を設定できます。この場合、ほかのコンテナが使用していないリソースがあれば、そのリソースを使用したいコンテナに割り当てられるようになります。最終的に、単一のホストで有効にできるコンテナ数は、使用可能な CPU 数とメモリー容量、および各コンテナで予約されているこれらのリソース量によって決まります。システムは、すべての有効なコンテナに必要なリソースの総量を備えている必要があります。リソース量は、アプリケーションのニーズによって決まります。

Container Manager でのコンテナの管理については、第 4 章を参照してください。

リソース管理

一般に、リソースは、プロセスにバインド可能な OS のエンティティを表します。通常は、カーネルサブシステムによって構築され、分割が可能なオブジェクトを表します。リソースは、アプリケーションの動作を変更するために操作可能な計算機システムの一部と考えることもできます。リソースの例には、物理メモリー、CPU、またはネットワーク帯域幅があります。

Container Manager は、Solaris 8、Solaris 9、および Solaris 10 のリソース管理ユーティリティと連動します。Solaris 8 では、Solaris Resource Manager 1.3 によってリソースが管理されます。各サービスは `lnode` で表されます。`lnode` を使用して、リソース割り当てポリシーとリソースの使用状況データが記録されます。`lnode` は、UNIX ユーザー ID (UID) に対応します。UID は、デフォルトで個々のユーザーとアプリケーションを表します。`lnodes` およびリソース管理の詳細は、『Solaris Resource Manager 1.3 のシステム管理』を参照してください。

Solaris 9 と Solaris 10 では、Resource Manager によってリソースが管理されます。このリリースでは、`lnode` に似ているものとしてプロジェクトがあります。プロジェクトは、関連する処理のネットワーク全体の管理 ID です。コンテナ内で実行されるすべてのプロセスはプロジェクト識別子 (プロジェクト *ID*) が同じです。Solaris カーネルによって、プロジェクト ID を使用してリソース使用状況が追跡されます。同じ追跡方法を使用する拡張アカウンティングを使用して履歴データを収集できます。Container Manager では、プロジェクトがコンテナに相当します。

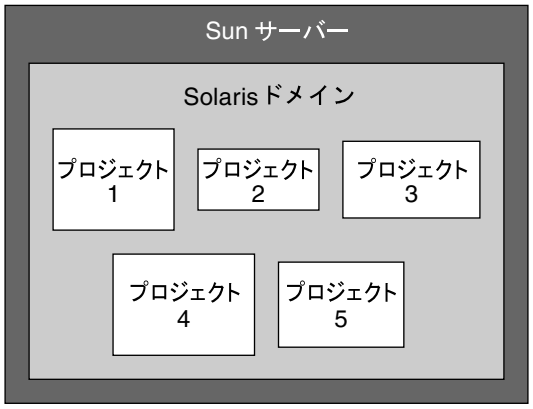


図 **1–3** ホスト上のプロジェクトの例

コンテナ内で実行されるプロセスに関する情報は、Container Manager の GUI で確認します。ソフトウェアを使用してコンテナを作成および管理すると、データは透過的に収集されます。

コンテナの境界は、複数の方法で作成できます。リソースプールを使用してシステムを分割する方法と、リソース制御によってプロジェクトに制限を設定する方法があります。

リソースプール

リソースプール (またはプール) は、Solaris 9 と Solaris 10 で、ホストのリソースを分割するために使用されているソフトウェア構成機構です。リソースセットは、プロセスにバインド可能なリソースです。リソースセットの例には、メモリーセットやプロセッサセットがあります。Solaris 9 と Solaris 10 では、プロセッサセットだけを使用できます。プールは、ホスト上で使用可能なさまざまなリソースセットをバインドしたものです。

リソースプールは、1 つまたは複数のプロジェクトを保持できます。プロジェクトが 1 つの場合、プールに関連付けられているリソースは、そのプロジェクト専用です。プロジェクトが複数の場合、プールに関連付けられているリソースは、プロジェクト間で共有されます。

Solaris 10 オペレーティングシステムでは、製品に動的リソースプールという機能があります。動的リソースプールを使用すると、システムイベントや負荷の変更に応じて各プールのリソース割り当てを調整できるので、パフォーマンスを向上できます。この機能については、28 ページの「動的リソースプール」を参照してください。

Solaris 8 オペレーティングシステムでは、ホストで使用できるリソースプールは 1 つだけです。このプールを `pool_default` といいます。この OS バージョンでは、リソースプールは存在しないので、見かけ上、`pool_default` が作成されます。Solaris 8 が動作するホスト上の CPU はすべて単一のプールに含まれると見なされます。

Container Manager によるリソースプールの管理については、第 5 章を参照してください。

リソースの制御

複数のプロジェクトが単一のプールにバインドされている場合は、単一のプロジェクトに保証 (制限) を設定できます。この制限をリソース制御といいます。制御の例には、公平配分スケジューラ (FSS) を使用する場合などの最小 CPU 制限の設定があります。また、`rcapd` デーモンを使用する場合などの物理メモリーキャップの設定もあります。最小 CPU 保証を設定すると、1 つのプロジェクトのアイドル状態の CPU サイクルを、ほかのプロジェクトのアプリケーションが使用できます。

ゾーン

ゾーンは、アプリケーションを実行するための独立した安全な環境です。ゾーンを使用すると、Solaris のインスタンス内で仮想化されたオペレーティングシステム環境を作成できます。ゾーンを使用すると、1 つまたは複数のプロセスを、システムのほかのプロセスから独立して実行できます。たとえば、ゾーン内で実行されているプロセスは、ユーザー ID やその他の資格情報に関係なく、同じゾーン内のほかのプロセスだけにシグナルを送信できます。エラーが発生した場合は、ゾーン内で実行されているプロセスだけに影響します。

大域ゾーン

すべての Solaris 10 システムには、OS の旧バージョンと同様に、大域ゾーンという一般的な大域環境が含まれます。大域ゾーンには 2 つの機能があります。大域ゾーンはシステムのデフォルトゾーンであり、またシステム全体の管理制御に使用されるゾーンでもあります。大域管理者が、非大域ゾーン (単に「ゾーン」ともいう) を作成していない場合、すべてのプロセスは大域ゾーン内で実行されます。

非大域ゾーンの構成、インストール、管理、およびアンインストールは、大域ゾーンからのみ行うことができます。システムハードウェアから起動できるのは、大域ゾーンだけです。物理デバイス、ルーティング、または動的再構成 (DR) などの管理作業は、大域ゾーンでのみ行うことができます。大域ゾーンで実行中の、適切な権限を持つプロセスやユーザーは、ほかのゾーンに関連付けられているオブジェクトにアクセスできます。

大域ゾーン内の権限がないプロセスやユーザーは、非大域ゾーン内の権限を持つプロセスやユーザーに許可されていない処理を行うことができる場合があります。たとえば、大域ゾーンのユーザーは、システムのすべてのプロセスに関する情報を表示できます。ゾーンを使用すると、管理者は、システム全体のセキュリティーを保ちながら、一部の管理作業を委任できます。

非大域ゾーン

非大域ゾーンには、専用の CPU、物理デバイス、物理メモリー領域が不要です。これらのリソースは、単一のドメインまたはシステム内で実行されている複数のゾーンで共有できます。ゾーンは、起動または再起動しても、システム上のほかのゾーンに影響ありません。各ゾーンの要件に応じて、カスタマイズしたサービスを提供できます。プロセスの基本的な独立のため、プロセスは同じゾーン内のプロセスだけを認識でき、また同じゾーン内のプロセスだけにシグナルを送信できます。ゾーン間の基本的な通信は、各ゾーンに 1 つ以上の論理ネットワークインタフェースを与えることで可能になっています。1 つのゾーン内で実行されているアプリケーションは、別のゾーンのネットワークトラフィックを確認できません (パケットのストリームは同じ物理インタフェースを通ります)。

ネットワーク接続が必要な各ゾーンには、1 つまたは複数の専用 IP アドレスを設定します。

ゾーンの詳細は、『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』を参照してください。

リソース使用状況レポートおよび拡張アカウンティングデータ

Performance Reporting Manager アドオン製品が Container Manager とともにインストールされている場合は、リソース使用状況の履歴データのレポートをコンテナ、リソースプール、ゾーン、プロジェクト、またはホストごとに作成できます。CPU データ、メモリー使用状況データ、および CPU の拡張アカウンティングデータは、Performance Reporting Manager のデータ収集サービスによってデータベースに保存されます。GUI から、リソース使用状況の詳細を示すグラフレポートを要求するか、コンマ区切り (CSV) 形式のテキストファイルにデータをエクスポートできます。テキストファイルは、たとえば請求や会計のアプリケーションで使用できます。

Performance Reporting Manager ソフトウェアの詳細は、『Sun Management Center 3.6 Performance Reporting Manager User's Guide』を参照してください。使用可能なレポートとアカウンティングデータについては、123 ページの「レポート」を参照してください。

Container Manager をご使用になる前に

Container Manager ソフトウェアをインストールし、使用する前に、リソース消費のニーズを評価します。コンテナ作成処理の一環として、コンテナ内で実行されるプロセスの最小 CPU 予約数と、必要に応じて物理メモリーキャップを指定します。ニーズを評価し、目標を定め、リソース計画が決定していると、コンテナの作成処理が簡単になります。また、作業の前にすべてのハードウェアの仕様のマスターリストも用意すると役立ちます。

サーバー統合

サーバーの統合を成功させるためには、統合の候補のすべてのサーバー、記憶装置、およびアプリケーションのマスターリストが重要です。統合計画が完成したら、このリストを使用して計画を実施します。

データセンターでサーバーを統合する場合は、Container Manager ソフトウェアをインストールし、使用する前に必要な作業があります。次の作業が必要です。

1. 統合するアプリケーションを選択します。
2. アプリケーションの作業負荷を構成するプロセス、ユーザーのグループ、ユーザーなどのコンポーネントを特定します。
3. 定義した各作業負荷のパフォーマンス要件を決定します。この作業では、CPU、メモリー、ネットワーク、および記憶装置の必要量と使用状況を含めて、現在のシステムでアプリケーションのリアルタイムのアクティビティを監視する必要があります。また、新しいシステムを構成し、読み取り専用ファイルシステム、ライブラリ、マニュアルページなどのリソースを効率よく共有するために各作業負荷が使用する、ファイルシステム、共有ファイルシステム、および共有ライブラリの種類を判断する必要があります。
4. どのアプリケーションにもっともリソースが必要か、およびリソースが必要な期間によって、システムリソースを共有する作業負荷に順位を付けます。同じシステム内の競合する作業負荷を特定する必要もあります。
5. これらの作業負荷のプロジェクトを特定します。プロジェクトは、関連する作業をグループにまとめるための管理名です。たとえば、Web サービスのプロジェクトや、データベースサービスのプロジェクトを設定できます。

注 – Solaris オペレーティングシステムでは数千個のコンテナを使用できますが、実際上は、またパフォーマンスのためには、200 個のホスト、ホスト当たり 10 ゾーン、ゾーン当たり 10 プロジェクトを推奨します。

サーバーの統合の計画および実行については、David Hornby、Ken Pepple 共著『Consolidation in the Data Center』(Sun Blueprints) を参照してください。

Container Manager 使用例

Container Manager の使用例を示します。

Oracle 用のゾーンがある複数のプロジェクト

この例では、ゾーンが設定されたデフォルトのリソースプールがあるとします。ここで、ゾーンが 2 つ、リソースプールが 1 つあるコンテナを 1 つ設定します。一方のゾーン zone_ora1 に Oracle データベースアプリケーション、他方のゾーン zone_ws01 に Web サーバーアプリケーションがあります。リソースプールにはそれぞれ 2 つの CPU があります。コンテナに 8 個の CPU シェア数を設定します。zone_ora1 用に 4 つ、zone_ws01 用に 3 つです。コンテナでは、公平配分スケジューラを使用します。

動的リソースプールの例

この例では、リソースプールが 2 つあるコンテナを 1 つ設定します。pool1 には、1 ～ 3 個の CPU が割り当てられます。pool1 の負荷の目標は、20 % より大きく、80 %未満です。pool2 は、メールサーバーで使用されます。他方のプールは動的なので、メールサーバーに必要な負荷によって、アプリケーションに 1 ～ 3 個の CPU を使用できます。

同一コンテナを共有するアプリケーション

この例では、ゾーンが 2 つあるコンテナを 1 つ設定します。最初のゾーン zone_ora02 には、7 つのプロジェクトがあります。ユーザー ORACLE 用のプロジェクト、データベース管理者のグループによって実行されるプロセス用のプロジェクト、および system、user.root、noproject、default、および group.staff の 5 つのデフォルトプロジェクトです。最初のゾーンには合計 100 個の CPU シェアがあります。デフォルトプロジェクトには、それぞれ 1 つのシェアを割り当てます。ユーザー ORACLE の最初のプロジェクトには 75 個のシェアを割り当て、group.dba 用の 2 つめのプロジェクトに 20 個のシェアを割り当てます。

2 つめのゾーン zone_ws_02 は Web サーバー用です。

複数のシステム上の Oracle 10g ラック

この例では、Oracle 10g アプリケーションを複数のシステムで実行します。Oracle 10g アプリケーション用に 1 つのプールと 1 つのゾーンがあるプロジェクトをシステム 1 に作成します。ゾーンとプールを含むプロジェクトを別のシステムにコピーし、そのシステムでプロジェクトを Oracle 10g アプリケーションに関連付けます。

複数のリソースプールがある複数のシステム

この例では、2 つのシステムがあり、それぞれに 2 つのプールがあります。Web サーバーがあるプロジェクトがシステム 1 に、Web サーバーがあるプロジェクトがシステム 2 にあります。各プロジェクトには 10 個の CPU シェアがあり、各 Web サーバーに 5 個のシェアを割り当てます。残りの 5 つのシェアは、将来のために残しておきます。

Solaris Container Manager 3.6 の新機能と変更点

Solaris Container Manager は、次に示す新機能を備えています。機能は、オペレーティングシステムによって異なります。

表 1–1 Solaris Container Manager の新機能

利点	機能	**Solaris 10 (SPARC および x86)**	**Solaris 9 (SPARC および x86)**	**Solaris 8 (SPARC)**
プロセスを独立して仮想 OS 環境で実行	ゾーン管理	あり		
システムパフォーマンスの目標の設定と取得	動的リソースプール	あり		
ネットワークの輻輳の回避	IPQoS (Internet Protocol Quality of Service)	あり		
より柔軟なプロセス管理	コンテナ間でのプロセスの移動	あり	あり	
タイムシェアスケジューラのサポート	ほかのスケジューラクラスのサポート	あり	あり	あり
仮想化ツールの改善	グラフの改善	あり	あり	あり
メモリーを割り当てるゾーンを認識するコンテナ	コンテナの改善	あり	あり	あり
上位 5 つのリソースオブジェクトの使用状況レポート	グラフの改善	あり	あり	あり

ゾーン管理

Container Manager を使用すると、非大域ゾーンを作成、削除、変更、停止、および再起動できます。Container Manager では、既存のゾーンの検出、ゾーンの変更の検出、ゾーンの CPU 、メモリー、およびネットワークの使用状況の監視とアーカイブ、およびゾーンの起動/停止アラームの生成も可能です。

ゾーンについては、第 6 章を参照してください。

動的リソースプール

動的リソースプールでは、設定されたシステムパフォーマンスの目標を達成できるように各リソースプールのリソース割り当てが動的に調整されます。動的リソースプールによって、システム管理者が決定する必要のある事項が減ります。調整は、システム管理者が指定したシステムパフォーマンスの目標を満たすように自動的に行われます。

動的リソースプールは、Solaris 10 システムで作成、変更、および削除できます。最小と最大の CPU 数、使用状況の目標、局所性の目標、CPU シェアなど、動的リソースプールの制約を設定したら、Container Manager のエージェントによって、リソースが使用可能かどうか、また消費の状況に合わせてプールのサイズが動的に調整されます。

リソースプールの設定は、エージェントとサービスの両方のデータベースに保存されます。

IPQoS を使用した帯域幅制御

IP Quality-of-Service 機能によって、ネットワークユーザーに一貫したレベルのサービスを提供し、またネットワークトラフィックを管理できます。このサービスを使用すると、ネットワークの統計情報の順位付け、制御、および収集が可能です。

この機能では、Solaris ゾーンのインバウンドトラフィックとアウトバウンドトラフィックが管理されます。ゾーンの入力/出力ネットワーク帯域幅の上限を指定します。制限を超えると、パッケージは破棄されます。IPQoS には、CPU オーバーヘッドが相当量あるので、これはオプションの機能です。

Container Manager によって、作業の使用状況データが監視および収集され、ネットワーク使用状況の履歴グラフが作成されます。

柔軟性のあるプロセス管理

プロセス管理の柔軟性を高めるため、Container Manager 3.6 ではプロセスをコンテナ間で移動できます。Solaris 9 システムでは、プロセスをコンテナ間で移動できます。Solaris 10 システムでは、同じゾーン内のコンテナ間だけでプロセスを移動できます。

タイムシェアスケジューラ

Container Manager 1.0 では、公平配分スケジューラ (FSS) だけを使用できました。Container Manager 3.6 では、リソースプールを作成または変更するときに、公平配分スケジューラクラスまたはタイムシェアスケジューラクラスのいずれかを選択することができます。スケジューラクラスによってプロセスの優先順位、つまり次に実行されるプロセスが決まります。

リソースプールのスケジューラクラスを変更したら、そのリソースプールの新しいプロセスが、リソースプールのスケジュールクラスに変更されます。Container Manager では、実行中のプロセスのスケジューラクラスは変更されません。

コンテナの強化機能

Container Manager には、コンテナに次の拡張機能があります。

- Solaris 10 では、コンテナはゾーンを認識します。各ゾーンには 5 つのデフォルトコンテナがあります。
- コンテナに特定の容量の共有メモリーを割り当てることができます。

Container Manager のマニュアル

この製品の関連マニュアルを次の表に示します。Solaris Container Manager 3.6 のマニュアルは、http://docs.sun.com/app/docs/coll/810.4 で入手できます。

表 **1–2** 関連マニュアル

作業	リソース
コンテナのインストールおよび管理	Solaris Container Manager 3.6 インストールと管理 (このマニュアル)
製品でヘルプの表示	Solaris Container Manager 3.6 オンラインヘルプ. このヘルプにアクセスするには、Solaris Container Manager の GUI で「ヘルプ」のリンクをクリックします。
Sun Management Center 3.6 およびアドオン製品のインストール(Container Manager を含む)	『Sun Management Center 3.6 インストールと構成ガイド』

表 1–2 関連マニュアル　(続き)

作業	リソース
インストールに関する問題、実行時の問題、最新情報 (サポート対象のハードウェアを含む)、およびマニュアルの問題に関する情報の確認	『Sun Management Center 3.6 ご使用にあたって』
Container Manager のオプションのアドオン Performance Reporting Manager に関する情報の確認	『Sun Management Center 3.6 Performance Reporting Manager User's Guide』
Solaris 8 オペレーティングシステムを使用している場合に、Solaris Resource Manager 1.3 に関する情報の確認	『Solaris Resource Manager 1.3 のインストール』 『Solaris Resource Manager 1.3 のシステム管理』 『Solaris Resource Manager 1.3 ご使用にあたって』
Solaris 9 または 10 オペレーティングシステムを使用している場合に、Solaris リソース管理とゾーンに関する情報の確認	『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』

基本的な使用方法

Solaris Container Manager をすでにインストールして設定している場合は、次のリンクをクリックすると、製品の操作方法を確認できます。

- 51 ページの「Container Manager の GUI を起動する」
- 73 ページの「プロジェクトの作成」
- 85 ページの「プロジェクト内でのプロセスの移動または起動」
- 88 ページの「プロジェクトの有効化と無効化」
- 97 ページの「リソース変更ジョブによるプロジェクトの変更」
- 104 ページの「新規リソースプールの作成」
- 110 ページの「非大域ゾーンの作成」
- 120 ページの「アラームのしきい値を設定する」
- 126 ページの「レポート要求」

第 2 章

Container Manager のインストールと設定

この章では、ウィザードを使用して Solaris Container Manager 3.6 (Container Manager) ソフトウェアのインストール、設定、およびアンインストールを行う手順を示します。コマンド行を使用してこれらの作業を行う方法については、付録 A を参照してください。

ソフトウェアのインストール、設定、および使用に関する最新の情報は、『Solaris Container Manager 3.6 リリースノート』を参照してください。

次の内容について説明します。

- 31 ページの「Container Manager ソフトウェアについて」
- 35 ページの「Container Manager ソフトウェアのインストール」
- 37 ページの「Container Manager ソフトウェアの設定」
- 41 ページの「バックアップと復元」
- 42 ページの「エージェント更新」
- 42 ページの「Container Manager ソフトウェアのアンインストール」

Container Manager ソフトウェアについて

Container Manager は、Sun Management Center 3.6 ソフトウェアの最新のパッチを適用したときにアドオン製品としてインストールされます。このアドオンソフトウェアは、ホストの機能によって、Sun Management Center のサーバーレイヤまたはエージェントレイヤにインストールされます。Sun Management Center は、サーバー、コンソール、およびエージェントの 3 つのレイヤから構成されるアプリケーションです。3 つのレイヤのアーキテクチャーについては、『Sun Management Center 3.6 インストールと構成ガイド』の「Sun Management Center の概要」を参照してください。

注 – Container Manager を使用するには、複数の追加ソフトウェア製品をインストールする必要があります。インストールを開始する前に、関連マニュアルをすべてお読みください。ほかの製品のインストールを完了するには、ソフトフェアパッチを適用したり、システムの再起動をしたりする必要がある場合があります。また、インストールを開始する前に、これらの製品に必要な容量を考慮する必要があります。

Sun Management Center と Solaris Container Manager の要件を次の表に示します。

必要なリソースの総量については、『Sun Management Center 3.6 インストールと構成ガイド』の付録 C「必要なハードウェアリソースの確認」を参照してください。

表 **2–1** Sun Management Center と Solaris Container Manager のシステム要件

ベースレイヤ	オペレーティングシステム	ディスク容量	**RAM**	スワップ領域
Sun Management Center サーバー (SPARC)	Solaris 8、Solaris 9、および Solaris 10 の Solaris Developer Software Group のインストール	合計 800M バイト /opt に 300M バイト /var/opt に 500M バイト	最小 512M バイト 小型～大型サーバーに 1G バイトを推奨 超大型サーバーに 2G バイトを推奨	1G バイトを推奨
Sun Management Center エージェント (SPARC)	Solaris 8、Solaris 9、および Solaris 10	エージェント当たり、/opt/SUNWsymon に 18M バイト エージェント当たり、/var/opt/SUNWsymon に 2M バイト	読み込まれるモジュールとシステムの種類に応じ、エージェント当たり 10 ～ 29M バイト	
Sun Management Center エージェント (x86)	Solaris 9 および Solaris 10	エージェント当たり、/opt/SUNWsymon に 18M バイト エージェント当たり、/var/opt/SUNWsymon に 2M バイト	読み込まれるモジュールとシステムの種類に応じ、エージェント当たり 10 ～ 29M バイト	

表 2–1 Sun Management Center と Solaris Container Manager のシステム要件　(続き)

ベースレイヤ	オペレーティングシステム	ディスク容量	RAM	スワップ領域
Solaris Container Manager サーバー (SPARC)	Solaris 8、Solaris 9、および Solaris 10	300M バイト	最小 512M バイト 小型～大型サーバーに 1G バイトを推奨 超大型サーバーに 2G バイトを推奨	1G バイトを推奨
Solaris Container Manager エージェント (SPARC および x86)	Solaris 8、Solaris 9、および Solaris 10 Solaris 8 で Solaris Container Manager を使用するには、Solaris Resource Manager 1.3 が必要です。Solaris 9 Update 5 以降では、共有メモリーがサポートされています。	エージェント当たり、/opt/SUNWsymon に 18M バイト エージェント当たり、/var/opt/SUNWsymon に 2M バイト	読み込まれるモジュールとシステムの種類に応じ、エージェント当たり 10 ～ 29M バイト	
Performance Reporting Manager サーバー (SPARC)	Solaris 8、Solaris 9、および Solaris 10	選択したレポートのオプションによって異なります。 ■ 小規模構成: 5G バイト ■ 中規模構成: 12G バイト ■ 大規模構成: 24G バイト	1G バイト	1G バイトを推奨
Performance Reporting Manager エージェント (SPARC および x86)	Solaris 8、Solaris 9、および Solaris 10	最小 8000K バイト 5 分間隔で 1000 個のプロパティを記録した場合は 80M バイト。		

Container Manager ソフトウェアは、Solaris オペレーティングシステムの次のバージョンで使用できます。

表 **2–2** Solaris OS のバージョン別の Container Manager の機能

OS バージョン	**Container Manager** の機能
Solaris 8 6/00 ～ Solaris 8 2/02 (SPARC® のみ)	Solaris Resource Manager 1.3 ソフトウェアに加えて、最小 CPU 予約数とメモリーキャップを指定できます。
Solaris 9 FCS ～ Solaris 9 8/03 (SPARC および x86)	最小 CPU 予約数を指定できます。メモリー管理はサポートされていません。
Solaris 9 12/03 以降 (SPARC および x86)	最小 CPU 予約数とメモリーキャップを指定できます。リソースプール管理もサポートされます。
Solaris 10 (SPARC および x86)	最小 CPU 予約数、メモリーキャップ、ソーン管理、動的リソースプール、IP サービス品質 (IPQoS) などの機能があります。

Container Manager エージェントは、Sun Management Center 3.6 でエージェントレイヤがサポートされているすべてのハードウェアプラットフォームで動作します。詳細は、『Sun Management Center 3.6 ご使用にあたって』の第 3 章「最新情報」を参照してください。

Container Manager ソフトウェアは、SPARC 版と x86 版のどちらも、構造上、中立的な次のソフトウェアパッケージから構成されます。

- Sun Management Center サーバー用のパッケージ: `SUNWscms`、 `SUNWscmc`、`SUNWscmca`、`SUNWscmdb`、 `SUNWscmh` (オンラインヘルプ)
- Sun Management Center のサーバーとエージェント用のパッケージ: `SUNWscmcm`、`SUNWscmp`
- Sun Management Center のエージェント用のパッケージ: `SUNWscma`

注 – Container Manager モジュールは、ソフトウェアのインストールと設定が完了したら自動的に読み込まれます。モジュールは、Sun Management Center の Java コンソールまたは Web コンソールに表示されません。ほかのアドオンソフトウェアと同様に、このモジュールを Sun Management Center で手動で読み込む必要はありません。

Container Manager ソフトウェアのインストール

注 – (Solaris 10 のみ) Container Manager は大域ゾーンにインストールします。

Container Manager のアドオンソフトウェアは、Sun Management Center 3.6 のインストールウィザードまたはコマンド行を使用してインストールできます。この節は、インストールウィザードを使用してインストールする方法について説明します。コマンド行によるインストールについては、付録 A を参照してください。

Container Manager ソフトウェアは、次のときにインストールできます。

- Sun Management Center 3.6 ソフトウェアと同時にインストールします。
- Sun Management Center 3.6 のインストール、または Sun Management Center 3.6 へのアップグレードの完了後。この場合、Container Manager ソフトウェアを別個にインストールします。

Container Manager のパフォーマンスとアカウンティングデータの機能を使用するには、Performance Reporting Manager ソフトウェアもインストールする必要があります。このソフトウェアのインストールと設定の詳細は、『Sun Management Center 3.6 Performance Reporting Manager User's Guide』を参照してください。これらの機能を使用しない場合は、このアドオンソフトウェアをインストールする必要はありません。

▼ Sun Management Center 3.6 へのアップグレード時に Container Manager をインストールする

手順

- アップグレードの手順の詳細は、『 **Sun Management Center 3.6** インストールと構成ガイド』を参照してください。**Performance Reporting Manager** などのほかのアドオンソフトウェアを同時にインストールすることもできます。

▼ Container Manager を別個にインストールする

手順

1. スーパーユーザー **(su -)** で、次のように入力して **Sun Management Center** のインストールウィザードを起動します。

   ```
   # /opt/SUNWsymon/sbin/es-guiinst
   ```

/opt は、Sun Management Center 3.6 がインストールされているディレクトリを示します。これとは異なるディレクトリの場合には、実際の名前に置き換えてください。

Sun Management Center のインストールウィザードが起動し、指示に従ってソフトウェアをインストールできます。「次へ」ボタンをクリックしてウィザードを進みます。

2. プロンプトが表示されたら、**Container Manager** ソフトウェアのファイルのソースディレクトリを指定します。
   - SPARC システムでは、次のいずれかを選択します。
     - ソフトウェア CD からインストールする場合は、次のように入力します。

       # **/<DiskMountDir>/image**
     - ソフトウェアがコピーされているディレクトリからインストールする場合は、次のように入力します。

       # *disk1***/image**

       disk1 は、ソフトウェアがコピーされた場所です。
   - x86 システムでは、次のいずれかを選択します。
     - ソフトウェア CD からインストールする場合は、次のように入力します。

       # **/<DiskMountDir>/x86/image**
     - ソフトウェアがコピーされているディレクトリからインストールする場合は、次のように入力します。

       # *disk1***/x86/image**

       disk1 は、ソフトウェアがコピーされた場所です。
3. マニュアルをインストールする場合は、「製品マニュアルのインストール」チェックボックスを選択します。

注 – この手順でインストールされるマニュアルは *Sun Management Center 3.6* コレクションです。Solaris Container Manager 3.6 のマニュアルは、このコレクションに含まれません。Solaris Container Manager 3.6 のマニュアルコレクションは、http://docs.sun.com/app/docs/coll/810.6 にあります。

インストールされたマニュアルを表示するには、ブラウザで /var/opt/sun_docs/sundocs.html にアクセスします。

4. サーバーレイヤにインストールする場合は、「言語サポートの選択」パネルで言語を選択します。このパネルは、サーバーレイヤにインストールする場合にのみ表示されます。

注 – この手順でインストールされる翻訳されたマニュアルは、*Sun Management Center 3.6* のマニュアルのみです。Solaris Container Manager 3.6 用の翻訳されたマニュアルは、http://docs.sun.com/app/docs/coll/810.6 にあります。

「使用可能な製品の検査」パネルが表示されます。進行状況の表示が終了すると、「インストール済みの製品」パネルが表示されます。

5. すでにインストールされている製品のリストを確認します。
   「アドオン製品の選択」パネルが表示されます。

6. インストールするアドオンソフトウェアのリストから **Solaris Container Manager 3.6** を選択します。
   「アドオン製品の許諾契約」パネルが表示されます。

7. 許諾契約を読みます。
   - 許諾契約に同意する場合は、「同意する」をクリックします。「確認」パネルが表示されます。
   - 許諾契約に同意しない場合は、「同意しない」をクリックします。インストールウィザードが終了し、インストールが終了します。

   インストールを続けるには、「同意する」ボタンをクリックします。

8. 「確認」パネルを確認します。
   ソフトウェアのインストールが完了すると、「インストール完了」パネルが表示されます。

   Container Manager ソフトウェアのインストールが完了したら、設定ウィザードの指示に従って、ソフトウェアの設定処理を実行できます。詳細は、37 ページの「Container Manager ソフトウェアの設定」を参照してください。

Container Manager ソフトウェアの設定

インストールが完了したら、Container Manager の設定ウィザードを実行して、サーバーレイヤとエージェントレイヤの設定を行います。ソフトウェアのインストールと同様に、設定処理は、Sun Management Center 3.6 のインストールまたはアップグレードの設定時に実行できます。あとで別個にソフトウェアを設定することもできます。

設定処理時には、次の場所に次の設定ファイルが作成されます。

- システムのプールとプロセッサセットの設定ファイルが `/etc/pooladm.conf` に作成されます。

- 拡張アカウンティングファイルが /var/sadm/exacct/*files* に作成されます。

/etc/pooladm.conf ファイルがすでにホストに存在する場合は、既存のファイルのバックアップが作成されます。このバックアップは、/etc/pooladm.conf.scmbak という名前になります。ホストに有効な設定がある場合は、リソースプールに関連付けられていないプロセッサセットがすべて削除されます。リソースプールのスケジューリングクラスは公平配分スケジューラ (FSS) に設定されます。最後に、新しい設定が確定され、/etc/pooladm.conf に保存されます。

ホストに有効な設定がない場合は、新しい設定が検出され、/etc/pooladm.conf に保存されます。検出された設定が確定されて有効になります。デフォルトのプールスケジューリングクラスが FSS に設定されます。

設定時にホストで拡張アカウンティングファイルが有効になっている場合は変更されません。ファイルが有効ではない場合は、設定され、有効になります。設定時に、拡張アカウンティングファイル名は、/var/adm/exacct/task*timestamp* に設定されます。

また、設定処理時にホストが次のように変更されます。

- メモリーキャップデーモン rcapd が開始されます (Solaris 8 OS、Solaris 9 12/03 OS、またはサポートされているリリース)。
- 対話式 (IA) スケジューリングクラスの下で実行されるプロセスがすべて FSS スケジューリングクラスの下に移動されます。
- タスクベースの拡張アカウンティングが有効になります。
- TS (タイムシェアスケジューラ) の指定がない場合は、ホストのスケジューリングクラスがデフォルトの FSS (公平配分スケジューラ) に設定されます。
- Solaris 8 オペレーティングシステムを使用しているシステムでは、lnode 階層がプロジェクト階層に従います。
- Solaris 8 オペレーティングシステムを使用しているシステムでは、group.staff プロジェクトが /etc/project ファイルから削除されます。

リソースプール、スケジューリングクラス、拡張アカウンティング、および Solaris のその他の標準リソース管理の概念については、『Solaris のシステム管理 (ネットワークサービス)』を参照してください。

注意 – Container Manager では、Solaris ソフトウェアのリソース管理の標準のコマンド行コマンドは使用できません。ソフトウェアの管理には、Container Manager の GUI を使用してください。

▼ Sun Management Center のインストールまたはアップグレード時に Container Manager を設定する

手順 ● 詳細および具体的な手順については、『**Sun Management Center 3.6** インストールと構成ガイド』の第 **6** 章「エージェントのインストールと更新、および **Microsoft Windows** へのインストール」を参照してください。

▼ Container Manager を別個に設定する

手順 **1.** スーパーユーザー **(su –)** で、次のように入力して **Sun Management Center** の設定ウィザードを起動します。

```
# /opt/SUNWsymon/sbin/es-guisetup
```

/opt は、Container Manager がインストールされているディレクトリを示します。これとは異なるディレクトリの場合には、実際の名前に置き換えてください。

Container Manager の設定ウィザードが表示され、指示に従ってソフトウェアの設定処理を実行できます。まず「概要」パネルが表示されます。「次へ」ボタンをクリックしてウィザードを進みます。前に戻って変更を行うには、「戻る」ボタンをクリックします。

注 – 複数の Sun Management Center のアドオン製品をインストールした場合は、自動的に各製品に対する設定ウィザードが表示されます。各ウィザードの指示に従って、アドオン製品の設定処理を実行します。1 つのアドオン製品の設定処理が完了すると、自動的に次の設定ウィザードが表示されます。Container Manager ソフトウェアの設定ウィザードが最初に表示されるとは限りません。

---

2. 「コンポーネントの停止」パネルのリストを確認します。

設定処理を続けるには、Sun Management Center のコンポーネントを停止する必要があります。インストールしたアドオン製品のいくつかの設定処理が完了すると、「拡張セットアップ」パネルが表示されます。残りのアドオンソフトウェアは、ここで設定できます。

3. 「アドオンの構成」オプションを選択します。

設定されていないインストール済みのアドオンソフトウェアを設定します。

4. 「セットアップ完了」パネル内の製品リストを確認します。

「アドオン製品の選択」パネルが表示されます。システムに新たにインストールされた設定対象のすべての製品が一覧表示されます。以前に設定した製品の設定を再度実行することもできます。

5. 製品リストに **Container Manager** が設定対象として表示されるかどうかを確認します。「次へ」をクリックして設定処理を開始します。

   コンポーネントの設定中に進行状況のパネルが表示されます。

6. エージェントレイヤを設定するときに、リソースプールに関連付けられていないプロセッサセットが検出された場合は、これらのセットを削除するためのアクセス権を要求するパネルが表示されます。次のオプションがあります。

   - 「はい」– これらのプロセッサセットはシステムから削除されます。Container Manager が正常に機能するには、リソースプールに関連付けられていないプロセッサセットはすべて削除する必要があります。プロジェクトはプールに関連付けられている必要があります。このオプションを選択したら、プロセッサセットは自動的に削除され、設定処理が続行します。
   - 「いいえ」– これらのプロセッサセットはシステムから削除されません。このオプションを選択したら、設定ウィザードが終了し、Container Manager ソフトウェアはシステムで設定されません。リソースプールに関連付けられていないプロセッサセットがあるホストでは、Container Manager ソフトウェアを使用できません。
   - 「取消し」– 設定ウィザードが終了します。Container Manager ソフトウェアはシステムで設定されません。

7. コンマで区切って有効なユーザー名を入力します。ここで入力するユーザーは、サーバーマシン上で **`/var/opt/SUNWsymon/cfg/esusers`** ファイルにすでに存在し、かつ **`esadm`** または **`esdomadm`** グループに割り当てられている必要があります。

   注 – Solaris Container Manager の設定ではユーザーの認証は行いません。

   ---

8. サーバーレイヤのインストールを設定する場合は、データベース用に **300M** バイト以上の空きディスク容量があるディレクトリのフルパスを指定し、「次へ」をクリックします。

9. 「**Solaris Container Manager DB** の構成」パネルの進行状況の表示が終了したら、「次へ」をクリックします。

10. 起動ウィザードを起動するには、「次へ」をクリックします。

    設定手順を完了すると、Sun Management Center のプロセスを開始できます。詳細は、『Sun Management Center 3.6 インストールと構成ガイド』の第 8 章「Sun Management Center の起動と停止」を参照してください。

    Container Manager の GUI の起動方法については、51 ページの「Container Manager の GUI を起動する」を参照してください。

プロファイルの作成

Solaris Container Manager の設定後、ユーザーはプロジェクト、プール、およびゾーンを管理するために次の作業を行う必要があります。

- ホストに必要なプロファイルを作成する
- ローカルアクセスユーザーリストにユーザーを加える

必要なプロファイルは次のとおりです。

- Solaris 10 の場合 - ゾーン管理、プール管理、プロジェクト管理
- Solaris 9 の場合 - プール管理、プロジェクト管理
- Solaris 8 の場合 - プロジェクト管理

次のコマンドを実行して、必要なプロファイルにユーザーが関連付けられていることを確認してください。

$ **profiles** <*username*>

必要なプロファイルがない場合は、次のコマンドをスーパーユーザーで実行し、ユーザーに関連付けられているプロファイルを変更します。

usermod -P "<必要なプロファイルをコンマで区切ったリスト>**"** <*username*>

システムにプロファイルが存在しない場合は、Solaris 管理コンソールまたは `smprofile` を使用して新しいプロファイルを作成します。コンマで区切ったプロファイルのリストが指定されない場合、指定した *username* の既存の設定が削除されます。

次のコマンドを実行して、ユーザーが SCM モジュールのローカルアクセスユーザーリストに含まれているかどうかを確認します。

<*BASEDIR*>**/SUNWsymon/sbin/es-config -M scm-container -s**

ユーザーがアクセスリストに含まれていない場合は、次のコマンドを実行します。

<*BASEDIR*>**/SUNWsymon/sbin/es-config -M scm-container -l** <*user_name*>

バックアップと復元

Sun Management Center のバックアップスクリプト (`es-backup`) と復元スクリプト (`es-restore`) を使用して、Container Manager のデータをバックアップまたは復元します。詳細は、『Sun Management Center 3.6 インストールと構成ガイド』の「Sun Management Center のバックアップと復元」を参照してください。

エージェント更新

Container Manager ソフトウェアの複数のホストのエージェントレイヤをアップグレードするには、エージェント更新機能を使用できます。エージェント更新機能を使用するときは、同じサーバーコンテキスト内の Container Manager エージェントマシンを同じ Sun Management Center バージョンにすることをお勧めします。サーバーコンテキストの詳細は、『Sun Management Center 3.6 ユーザーガイド』の「Sun Management Center のアーキテクチャー」を参照してください。

インストール時のエージェント更新機能の使用については、『Sun Management Center 3.6 インストールと構成ガイド』の「エージェントのインストールおよび更新イメージの作成」を参照してください。

Container Manager ソフトウェアのアンインストール

Container Manager ソフトウェアのアンインストールは、Sun Management Center の標準の手順で行います。コマンド行またはアンインストールウィザードのいずれかを使用できます。ソフトウェアパッケージに加えて、`base-modules-d.dat` のモジュールエントリと、Sun Web Console のアプリケーション登録が削除されます。削除処理中に、データベースに保存されているデータを残すか、削除するかを選択できます。インストール時に行ったシステムリソースの設定の変更は、Container Manager ソフトウェアのアンインストール時には削除されません。

コマンド行を使用したソフトウェアのアンインストールについては、付録 A を参照してください。標準的な Sun Management Center の削除手順は、『Sun Management Center 3.6 インストールと構成ガイド』の付録 A「SyMON とSun Management Center のアンインストール」を参照してください。

▼ Container Manager ソフトウェアを削除する

手順 **1.** スーパーユーザー **(su –)** で、次のように入力してアンインストールウィザードを起動します。

```
# /opt/SUNWsymon/sbin/es-guiuninst
```

`/opt` は、Sun Management Center がインストールされているディレクトリを示します。これとは異なるディレクトリの場合には、実際の名前に置き換えてください。Sun Management Center のアンインストールウィザードが表示されます。

2. ソフトウェアのリストから **Container Manager** を選択し、「次へ」をクリックします。

3. データファイルの保存場所を指定して、「次へ」をクリックします。

   ソフトウェアのアンインストールが開始します。Container Manager のパッケージと設定ファイルが削除されます。データファイルを残さないことを選択した場合は、データファイルも削除されます。

4. 「閉じる」をクリックしてウィザードを終了します。

第 3 章

コンテナの概要と製品の起動

この章では、コンテナとプロジェクトについて、また製品の起動方法について説明します。

この章では、以下の内容について説明します。

- 45 ページの「コンテナ の概要」
- 47 ページの「コンテナのプロパティー」
- 49 ページの「プロジェクトの状態」
- 50 ページの「Container Manager の GUI」
- 59 ページの「デフォルトのコンテナ」
- 61 ページの「コンテナの作成」

コンテナ の概要

プロジェクトは、ホストに関連付けられているコンテナです。プロジェクトは、一連の物理システムリソースの構成と管理に使用します。プロジェクトは、全体的なサーバー統合計画を実行するときに役立ちます。プロジェクトには、次の機能があります。

- アプリケーションの最小 CPU 予約数とメモリーキャップの設定によるシステムリソースの分散

  各アプリケーションは、プロジェクトに設定された最小 CPU 予約数とメモリーキャップ (省略可能) が保証されます。たとえば、アプリケーションベースのプロジェクトでは、ネットワーク上で実行される任意のアプリケーションに最小 CPU 予約数とメモリーキャップの両方を設定できます。同じホストで複数のプロジェクトが有効で、アプリケーションごとにプロジェクトがある場合、システムリソースの競合が減ります。この場合、そのホストで動作する Solaris カーネルによって、各プロジェクトに設定されているリソース境界が適用されます。
- データセンター全体でのリソース使用状況の追跡

プロジェクトのメンバーを使用して、プロジェクトで使用されているリソースの総量が追跡されます。UNIX ユーザーやグループは複数のプロジェクトに割り当てることができますが、ユーザーはデフォルトプロジェクトに割り当てられている必要があります。UNIX ユーザーが開始するプロセスは、ユーザーがメンバーである任意のプロジェクトにバインドできます。プロジェクトでは、プロジェクトのメンバーとリソースプールの両方を使用して、アプリケーションのリソース消費が追跡および制御されます。

プロジェクトの働きと、プロジェクトが使用するリソース管理ツールの詳細は、第 1 章を参照してください。

- 共有メモリーの割り当て

ソフトウェアをインストールし、設定したら、すぐに使用できるデフォルトプロジェクトがいくつかあります。ウィザードに従って独自のプロジェクトを作成することもできます。すべてのプロジェクトはコンテナに関連付けられています。このコンテナは、新規プロジェクトの作成に繰り返し使用できます。プロジェクトには、次の利点があります。

- CPU セットとメモリーのリソース境界が同じ複数のプロジェクトインスタンスを作成できる。
- 複数のホスト間でプロジェクトインスタンスを分散できる。1 つの有効なプロジェクトを複数のホストに分散することはできませんが、追加で作成した、同じコンテナのプロジェクトインスタンスは、 別のホストに割り当てることができます。
- 新しいプロジェクトインスタンスを短時間で作成できるので、必要なときに簡単にリソースを配分できる。

GUI はブラウザに表示され、3 つの表示 (タブ) があります。ホストの観点からの表示、コンテナの観点からの表示、および未確認のアラームの表示です。 ホストの表示とコンテナの表示は、グループを作成し、グループに含める要素を選択することで、構成を変更できます。

また、ソフトウェアを使用し、コンテナ内で実行されているプロセスと使用されているリソースを簡単に確認できます。コンテナまたはホストごとのリソース使用状況を評価するのに役立つグラフのオプションもいくつかあります。データをファイルにエクスポートすることもできます。これらの機能を使用すると、リソース消費を監視および再評価し、適切に調整できます。

ソフトウェアのアラーム機能を使用すると、コンテナのリソース消費が設定したしきい値に達したときに電子メールの通知を受け取ることができます。GUI では、ホストとコンテナにアラームのアイコンが表示されます。

リソース変更ジョブ機能を使用すると、1 回の要求で、1 つまたは複数のコンテナの現在のリソース境界を変更するスケジュールを 設定できます。リソース変更ジョブの作成または変更は、ウィザードに従って行うことができます。

コンテナのプロパティー

コンテナには、次のプロパティーがあります

- 名前
- 説明 (省略可能)
- プロジェクトの種類
- コンテナに関連付けられているプロジェクト名
- コンテナアプリケーションに関連付けられている UNIX ユーザーとグループ
- 一致式 (存在する場合)

コンテナに割り当てる名前は変更できません。プロジェクト名も同様です。コンテナのその他の識別情報は変更できます。

コンテナはソフトウェアによって保存され、コンテナが削除されるまで繰り返し使用できます。プロジェクトは、ホストに関連付けられているコンテナです。プロジェクトは、ホストに関連付け、リソース予約を設定したときに有効になります。

定義とリソース予約が同じ複数のプロジェクトを複数の異なるホストで同時に有効にできるので、コンテナを使用してデータセンター全体のプロジェクトを簡単に管理できます。コンテナを保存したら、そのコンテナを使用して、任意のホストでプロジェクトを有効にできます。つまり、コンテナは、新規プロジェクト作成用のテンプレートとして使用できます。

コンテナは、複数のプロジェクトのテンプレートとして機能します。コンテナでは、プロジェクトの共通のプロパティが 1 個所に保存されます。プロジェクトの共通のプロパティーは、次のとおりです。

- プロジェクト名。
- プロジェクトに含めるプロセスを判別する方法 (たとえば、コンテナに関連付けられている UNIX ユーザーやグループ、または一致式)

CPU シェアやメモリー制限などのその他のプロパティは、プロジェクトが有効になっているホストに固有です。Solaris Container Manager 3.6 では、この一連の共通のプロパティーは、1 個所に保存され、コンテナと呼ばれます。コンテナが特定のホストで有効になると、Solaris プロジェクトとしてインスタンス化され、/etc/project に保存されます。

たとえば、電子メールアプリケーション用のコンテナを設定するとします。プロジェクトの共通のプロパティーは、次のとおりです。

- プロジェクト名: mail_services。
- プロジェクトに含めるプロセスを判別する方法:「mozilla」を含む一致式。

コンテナが特定のホストで有効になると、プロジェクトをインスタンス化し、リソースプール、CPU シェア数、およびメモリー制限を指定できるようになります。

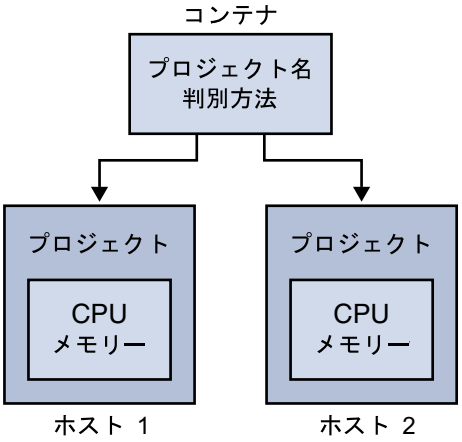


図 **3–1** コンテナとプロジェクト

コンテナを使用して、複数のゾーンやホストに複数のプロジェクトを作成できます。たとえば、単一のコンテナを使用して、3 つの異なるホストに 3 つの有効なプロジェクトを作成した場合、1 つのコンテナと、そのコンテナに 3 つのプロジェクトができます。コンテナの情報を変更すると、そのコンテナに基づいているすべてのプロジェクトが変更されます。

新規プロジェクトウィザードでは、すべての手順を完了したときに有効になるプロジェクトを作成できます。同時にコンテナが作成され、その名前が GUI に保存されます。コンテナだけを作成し、ウィザードに従ってプロジェクトをあとで有効にすることもできます。

コンテナは、GUI を使用して次の操作を行うことができます。

- 新規コンテナの作成
- 新規グループの作成
- コンテナのグループへの移動、またはグループからの移動
- プロパティの変更
- ホストへの新規プロジェクトの作成
- コンテナの削除

プロジェクトは、GUI を使用して次の操作を行うことができます。

- 新規グループの作成
- プロジェクトの移動
- プロパティの変更
- リソース予約の変更
- リソース変更ジョブの作成
- 内部で実行中のプロセスの状態の確認
- 使用状況レポートの要求とデータのファイルへのエクスポート
- プロジェクトの無効化または有効化

- アラームの設定
- プロジェクトの削除

プロジェクトの状態

アプリケーションに設定したリソース消費の境界は、実際にはプロジェクトによって適用されるのではありません。最小 CPU 予約数とメモリーキャップを指定し、プロジェクトが有効になったら、Solaris カーネルによってこれらの境界が適用されます。プロジェクトを使用する前に、プロジェクトの状態を理解する必要があります。プロジェクトには、定義済み、有効、および無効の 3 つの状態があります。

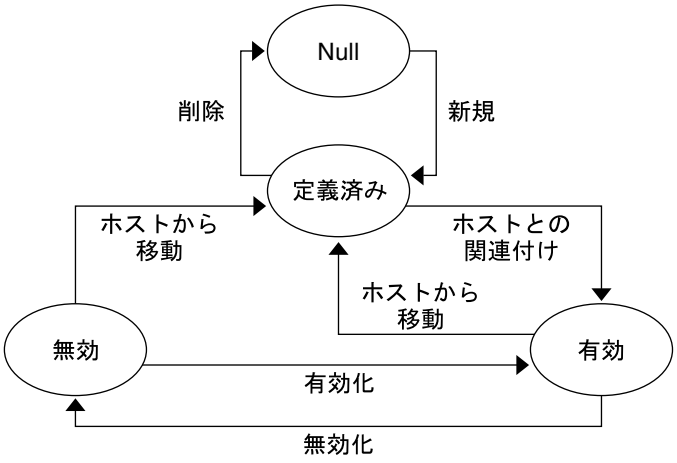


図 **3–2** プロジェクトの状態

プロジェクトは、その存続期間中、これらの状態の間を行き来します。

コンテナとプロジェクト

コンテナは、プロジェクト自体がまだ発生していない初期の段階で作成します。各プロジェクトには固有の名前が必要で、各プロジェクトは無限にデータベースに保存できます。

図 3–2 に、コンテナがホストに関連付けられたあとでプロジェクトが有効な状態になるところを示しています。プロジェクトは、無効にし、ホストとの関連付けを解除したら、定義済みの状態に戻すことができます。

プロジェクトの有効化

プロジェクトを有効にするには、まずコンテナをホストに関連付けます。次に、プロジェクトの最小 CPU 予約数とメモリーキャップを割り当ててリソース境界を設定します。プロジェクトは、これらのリソース境界をサポートするホストに関連付ける必要があります。有効なプロジェクトとは、外に出されてホスト上に常駐しているという点から、配備済みとも言えます。

新規プロジェクトウィザードでアプリケーションベースのプロジェクトを作成するときは、アプリケーションに関連するプロジェクトを特定する一致式を指定できます。一致式と一致するプロセスがすべてこのコンテナの下に移動されます。プロジェクトを有効にすると、コンテナが関連付けられているホストで、/etc/project データベースにエントリが作成されます。これに対して、一致するプロセスがコンテナのプロジェクト名の下に移動します。プロセスが移動したら、プロジェクトのすべてのリソース使用状況データが収集され、保存されます。

プロジェクトの無効化

プロジェクトを無効化すると、リソース境界が適用されなくなります。無効にしたプロジェクトは無効な状態になり、ホストの `/etc/project` ファイルから削除されます。無効なプロジェクトはソフトウェアのデータベースに残っているので、再度有効にできます。無効なプロジェクトを再び有効にすると、コンテナのリソース境界が再び適用されます。

プロジェクトが有効であった間に収集された使用状況データは、すべてデータベースに保存されます。プロジェクトを無効にしてから 30 日以内は、そのプロジェクトの使用状況レポートを要求できます。

Container Manager の GUI

Container Manager ソフトウェアでは、Solaris ソフトウェアのリソース管理の標準のコマンド行コマンドは使用できません。コンテナは、Container Manager のグラフィカルユーザーインタフェース (GUI) を使用して管理する必要があります。GUI は、ブラウザを使用して Java Web Console から起動します。サポートされているブラウザは次のとおりです。

- Netscape Navigator™ 4.7、6.2x、および 7.x (Solaris 8、9)
- Netscape Navigator 4.7x、6.2x、および 7.x (Microsoft Windows 98/2000/XP)
- Mozilla™ 1.4 以降
- Internet Explorer 5.0 以降 (6.x を含む) (Microsoft Windows 98/2000/XP)

▼ Container Manager の GUI を起動する

手順 **1.** **`/var/opt/SUNWsymon/cfg/esusers`** ファイルに **UNIX** ユーザー **ID** がない場合は、エントリを作成します。

ユーザーは、`esadm` または `esdomadm` のいずれかのグループに割り当てる必要があります。

エントリの作成とグループへの割り当ての手順については、「 Setting Up Users」in 『Sun Management Center 3.6 Installation and Configuration Guide』を参照してください。

2. ブラウザを起動します。

サポートされているブラウザの一覧は、50 ページの「Container Manager の GUI」を参照してください。

3. 次のように入力して **Container Manager** の **GUI** を表示します。

`https://`*sunmc-server_machine_name*`:6789/containers`**

Java Web Console のログインページが表示されます。

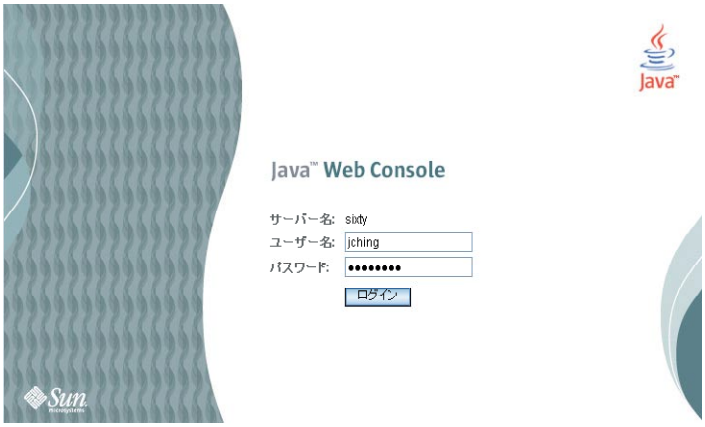


図 **3–3** Java Web Console のログインページ

ログインページが表示されない場合は、Java Web Console を再起動する必要がある可能性があります。再起動の方法については、52 ページの「Java Web Console を再起動する」を参照してください。

ヒント – 「コンソール」ページが表示されたら、「システム」セクションの下にある Solaris Container Manager 3.6 のリンクをクリックして GUI を表示します。

---

4. **UNIX** ユーザー **ID** とパスワードを使用して **Java Web Console** にログインします。

   Container Manager の GUI が表示されます。画面には、「ホスト」、「コンテナ」、および「未確認のアラーム」の 3 つのタブがあります。

図 **3–4** Container Manager のメインページ

▼ Java Web Console を再起動する

Java Web Console にアクセスできない場合は、このコマンドを使用して再起動します。

手順

- スーパーユーザー **(su -)** で、次のように入力して **Java Web Console** を再起動します。

```
# /usr/sbin/smcwebserver restart
```

Container Manager の GUI のタブ

Container Manager GUI の右側の区画に表示されるタブの説明を次の表に示します。

表 **3–1** Container Manager の GUI のタブ

タブ	タブ名	内容
ホスト (表示)	内容	選択したホスト上のリソースプールに関する情報が表示されます。
	プロパティ	選択したホスト、ゾーン、プロジェクト、またはリソースプールに関する情報が表示されます。
	使用状況	ホスト、ゾーン、プロジェクト、またはプールの日別、週別、または月別のリソース使用状況に関する情報が表示されます。有効なプロジェクトについては、リアルタイムの使用状況データが表示されます。このタブは、Performance Reporting Manager ソフトウェアがインストールされている場合にのみ使用できます。
	プロジェクト	ホストに関連付けられているプロジェクトに関する情報が表示されます。
	ゾーン	ホストに関連付けられているゾーンに関する情報が表示されます。
コンテナ (表示)	内容	プロジェクトに関する情報が表示されます。
	プロパティ	選択したホスト、コンテナ、プロジェクト、またはリソースプールに関する情報が表示されます。
	使用状況	ホスト、ゾーン、プロジェクト、またはプールの日別、週別、または月別のリソース使用状況に関する情報が表示されます。有効なプロジェクトについては、リアルタイムの使用状況データが表示されます。このタブは、Performance Reporting Manager ソフトウェアがインストールされている場合にのみ使用できます。
	ジョブ (リソース変更ジョブ)	予定されているリソース変更ジョブに関する情報が表示されます。このタブで新規リソース変更ジョブを作成することもできます。デフォルトコンテナにリソース変更ジョブを関連付けることはできません。
未確認のアラーム		重要度、メッセージ、管理対象オブジェクト、開始時刻、確認など、未確認のアラームに関する情報が表示されます。
リソースプール (下の階層)	内容	選択したリソースプールのゾーンに関する情報が表示されます。

表 **3–1** Container Manager の GUI のタブ　*(続き)*

タブ	タブ名	内容
	プロパティ	選択したリソースプールのプロパティに関する情報が表示されます。
	使用状況	プールの日別、週別、または月別のリソース使用状況に関する情報が表示されます。このタブは、Performance Reporting Manager ソフトウェアがインストールされている場合にのみ使用できます。
	プロジェクト	選択したリソースプールに関連付けられているプロジェクトに関する情報が表示されます。
ゾーン (下の階層)	内容	選択したゾーンのプロジェクトに関する情報が表示されます。
	プロパティ	選択したゾーンのプロパティに関する情報が表示されます。
	使用状況	ゾーンの日別、週別、または月別のリソース使用状況に関する情報が表示されます。このタブは、Performance Reporting Manager ソフトウェアがインストールされている場合にのみ使用できます。
プロジェクト (下の階層)	プロパティ	選択したプロジェクトのプロパティに関する情報が表示されます。
	使用状況	プロジェクトの日別、週別、または月別のリソース使用状況に関する情報が表示されます。このタブは、Performance Reporting Manager ソフトウェアがインストールされている場合にのみ使用できます。
	プロセス	選択したプロジェクトのプロセスに関する情報が表示されます。
	アラームしきい値	アラームしきい値を設定または削除できます。

ホスト表示

ホスト表示では、ホストの観点から情報が構成されます。管理しているエージェントマシンがすべてナビゲーションウィンドウに表示されます。ホスト名の横の三角形をクリックすると、各ホストで使用可能なリソースプールが表示されます。この表示では、ホストに関連付けられているコンテナを管理することもできます。

ソフトウェアがインストールされているエージェントのホストは、すべて自動的に検出され、ホスト表示に追加されます。この表示は、ナビゲーションウィンドウの左側のタブから表示します。検出されたエージェントのホストは、最初は「ホスト」というデフォルトのグループの下に配置されます。新規グループを作成し、関連するグループにホストを移動することで、このビューの構成を変更できます。

注 – Sun Management Center のサーバーコンテキストに含まれ、かつ Solaris Container Manager 3.6 がインストールされているエージェントマシンだけがホスト表示に読み込まれます。サーバーコンテキストの詳細については、『Sun Management Center 3.6 ユーザーガイド』の「Sun Management Center のアーキテクチャー」を参照してください。

ホスト表示のタブと情報を表 3–1 に示します。

ホストに関連付けられているプロジェクトインスタンスに関する情報は、「プロジェクト」表に表示されます。

デフォルトプールに関連付けられている「プロジェクト」表が表示されている「ホスト」表示を次の図に示します。

図 **3–5** 例:「プロジェクト」表が表示された「ホスト」表示

「プロジェクト」表には、各プロジェクトに関する情報が表示されます。1 行に 1 つのプロジェクトが表示されます。表示される情報は、次のとおりです。

プロジェクト名　　プロジェクトの名前

コンテナ名	コンテナの名前
状態	プロジェクトの状態 (有効または無効)
リソースプール名	プロジェクトがバインドされているリソースプール
ゾーン名	プロジェクトが常駐するゾーンの名前。Solaris 8 と Solaris 9 のホストでは、ゾーン名は常に「global」です。
CPU 予約数 (CPU シェア数)	プロジェクトの最小 CPU シェア数
CPU 使用量 (CPU 数)	プロジェクトで使用されている CPU 数
メモリーキャップ (M バイト)	メモリーの上限 (M バイト単位)
メモリー使用量 (M バイト)	プロジェクトで使用されているメモリー (M バイト単位)
共有メモリー (M バイト)	プロジェクト内で実行されるプロセスに使用できるメモリーの総容量 (M バイト単位)

「リソースプール」表には、各リソースプールに関する情報が表示されます。表示される情報は、次のとおりです。

リソースプール名	リソースプールの名前です。
現在の CPU 数	リソースプールに現在設定されている CPU 数
予約されていない CPU シェア数	リソースプール内で、ゾーンまたはプロジェクトに割り当てられていない CPU シェア数
スケジューラ	リソースプールに設定されているスケジューラ (タイムシェアスケジューラまたは公平配分スケジューラ)
CPU シェア数	リソースプールに設定されている CPU シェア数
最小 CPU 予約数	リソースプールに設定されている最小 CPU 数
最大 CPU 予約数	リソースプールに設定されている最大 CPU 数

「ゾーン」表には、各ゾーンに関する情報が表示されます。表示される情報は、次のとおりです。

ゾーン名	ゾーンの名前です。
ゾーンの状態	ゾーンの状態 (構成済み、不完全、インストール済み、準備完了、稼働、停止、またはダウン)
ゾーンのホスト名	仮想ホストとしてのゾーンの一意の名前
ゾーンのパス	ルート (/) ディレクトリを基準にした絶対パス
IP アドレス	ゾーンの IP アドレス

プロジェクトの CPU シェア数	ゾーン内のプロジェクトに割り当てられている CPU シェア数
予約されていない CPU シェア数	ゾーンに関連付けられているプロジェクトに割り当て可能な CPU シェア数
予約済み CPU シェア数	リソースプール内でこのゾーンに割り当てられている CPU シェア数
リソースプール	ゾーンのリソースプール

コンテナ表示

コンテナ表示では、コンテナの観点から情報が構成されます。コンテナとプロジェクトがすべてナビゲーションウィンドウに表示されます。コンテナは新規プロジェクトの作成に繰り返し使用できるので、このビューではコンテナに簡単にアクセスでき、またその他の管理作業を行うことができます。

インストールと設定が完了したら、コンテナ表示に「コンテナ」グループがデフォルトで自動的に追加されます。コンテナはコンテナ表示で管理します。

コンテナ表示を次の図に示します。

図 **3–6** 例: デフォルトコンテナに関連付けられているホストが表示された「コンテナ」表示

コンテナ表示に[表示される情報は、表 3–1 にあります。

ホストとコンテナのグループ化

ホスト表示には、デフォルトのグループ「ホスト」があります。ソフトウェアのインストール後に検出されたホストはすべてこのグループに配置されます。同様に、コンテナ表示にはデフォルトのグループ「default」があり、ホストのデフォルトコンテナがすべてこのグループに配置されます。これらのビューでは、ホストやコンテナを構成する追加グループを作成できます。

グループを使用して、データセンター内の 10 個のシステムでも、100 個のシステムでも構成できます。たとえば、同じ場所にあるホストを 1 つのグループに入れることができます。また、同じ顧客 (内部または外部) または部署が所有するコンテナを 1 つのグループに入れることができます。同様に、似ているアプリケーションのコンテナを 1 つのグループに入れることができます。

▼ コンテナグループまたはホストグループを作成する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで表示を選択します。
   - 新規コンテナグループを作成するには、コンテナ表示を選択します。右側の区画に「コンテナ」表が表示されます。
   - 新規ホストグループを作成するには、ホスト表示を選択します。右側の区画に「ホストとグループ」表が表示されます。
3. 「新規グループ」ボタンをクリックします。
   ダイアログが表示されます。
4. グループの名前を入力し、「了解」をクリックします。
   名前は 32 文字を超えてはいけません。

   新しいグループが選択した表示で表示されます。

▼ コンテナまたはホストを別のグループに移動する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで表示を選択します。
   - コンテナを別のグループに移動するには、コンテナ表示を選択します。右側の区画に「コンテナ」表が表示されます。

- ホストを別のグループに移動するには、ホスト表示を選択します。右側の区画に「ホストとグループ」表が表示されます。

3. 表の「移動」ボタンを使用可能にするには、移動するコンテナまたはホストのチェックボックスを選択します。
4. 右側の区画で、「移動」ボタンをクリックします。
   使用可能なグループがダイアログに表示されます。
5. コンテナまたはホストを移動するグループを選択します。
6. 「了解」をクリックします。
   コンテナまたはホストが、選択したグループに移動します。

デフォルトのコンテナ

ソフトウェアを設定したら、コンテナ表示に、「Default」というグループが読み込まれます。Solaris 9 または Solaris 10 オペレーティングシステム (OS) のホストでは、このグループに次の 5 つのデフォルトコンテナがあります。

- Default
- Processes with No Project
- Root User
- System Processes
- Users with Group Staff

各デフォルトコンテナは、`/etc/project` ファイル内に対応するエントリがあります。`default`、`noproject`、`user.root`、`system`、および `group.staff` です。

注 – Solaris 8 のホストでは、Users with Group Staff (group.staff) のコンテナは存在しません。その他のデフォルトコンテナは同じです。

図 **3–7** 例:「System」コンテナグループのコンテナの表示

各デフォルトコンテナは有効な状態で、境界は最小 CPU 予約数 (CPU シェア数) 1とメモリーキャップなしに設定されています。デフォルトコンテナは、必ずホストのデフォルトのリソースプール (`pool_default`) にバインドされます。Performance Reporting Manager がインストールされている場合は、リソース使用状況を監視し、各デフォルトコンテナのレポートを実行できます。

デフォルトコンテナは、無効化、編集、または削除できません。したがって、各デフォルトコンテナには「読み取り専用」と表示されています。

すべての UNIX ユーザーはデフォルトプロジェクトに割り当てられ、その結果、デフォルトコンテナに割り当てられます最初は、システムで実行されているすべてのプロセスがデフォルトコンテナに保持されます。プロジェクトを作成すると、プロセスは、対応するデフォルトコンテナから、作成したプロジェクトに移動されます。

コンテナの作成

プロジェクトはコンテナに基づきます。プロジェクトには 3 種類があります。種類は作成時に選択します。プロジェクトの種類によって、プロセスの追跡方法が決まります。

プロジェクトの種類

新規コンテナを作成するときは、プロジェクトの種類を選択する必要があります。プロジェクトは、ネットワーク全体で関連する作業を管理するための識別子 (ID) です。コンテナ内で実行されるプロセスはすべてプロジェクト ID が同じで、コンテナによってプロジェクト ID で使用されるリソースが追跡されます。コンテナの種類は、コンテナの作成時に選択します。

すべてのコンテナには、変更できないプロジェクト名があります。ホストでコンテナが有効になると、そのホストの `/etc/project` ファイルにこのプロジェクト名が追加されます。このエントリは、そのホストでコンテナが有効である間、維持されます。

1 つのホストで同じプロジェクト名のプロジェクトを同時に 2 つ有効にすることはできません。コンテナ内で実行されるプロセスはプロジェクト ID で追跡されるので、ホスト上のプロジェクト名はすべて固有である必要があります。

ユーザーベースまたはグループベースのプロジェクトを作成するとき、ユーザー名またはグループ名がプロジェクト名の一部に使用されます。ユーザーベースのコンテナの場合、プロジェクト名は user.*username* の書式で指定します。グループベースのコンテナの場合、プロジェクト名は group.*groupname* になります。したがって、ユーザーベースまたはグループベースのプロジェクトを作成するときは、`/etc/project` ファイルのデフォルトコンテナのエントリと重複するユーザー名またはグループ名を使用できません。詳細は、59 ページの「デフォルトのコンテナ」を参照してください。

アプリケーションベースのコンテナを作成するときは、任意のプロジェクト名を指定します。新規プロジェクトウィザードでは、異なるアプリケーションベースのプロジェクトに重複するプロジェクト名を指定できます。ただし、プロジェクト名が同じ 2 つのアプリケーションベースのプロジェクトを同じホストで同時に有効にすることはできません。コンテナを別のホストで有効にする場合にのみプロジェクト名を再利用してください。同じプロジェクト名のプロジェクトがあるホストでプロジェクトを有効にしようとすると、有効化に失敗します。

3 つのプロジェクトの種類の詳細と、選択したときの変更内容を次の表に示します。

表 **3–2** プロジェクトの種類の詳細

プロジェクトの種類	**OS** バージョン	詳細
ユーザーベース	Solaris 8	Solaris 8 でサポートされている唯一のプロジェクトの種類です。 `/etc/project` ファイルのプロジェクト名は user.*username* になります。プロジェクトはユーザーの一次デフォルトプロジェクトになります。
	Solaris 9 および Solaris 10	`/etc/project` ファイルのプロジェクト名は user.*username* になり、このプロジェクトに後で加わることができる UNIX ユーザーの一覧が指定されます。 有効な書式は *username* です。
グループベース	Solaris 9 および Solaris 10	`/etc/project` ファイルのプロジェクト名は group.*groupname* になります。 有効な書式は *groupname* です。
アプリケーションベース	Solaris 9 および Solaris 10	プロジェクト名には、アプリケーション名またはその他の任意の名前を使用できます。指定した名前が `/etc/project` ファイルに追加されます。 一致式を指定して、一致するプロセスをプロジェクト名に自動的に移動できます。一致式では、大小文字が区別されます。 プロセスの実行に使用される、対応する *username* または *groupname* を指定する必要があります。

リソース予約 (CPU シェア数)

プロジェクトを使用してアプリケーションのリソース管理を開始する前に、アプリケーションのリソースの傾向を把握する必要があります。ORACLE® など、特定のアプリケーションは、メモリーキャップの容量が不足すると、パフォーマンスが低下します。すべてのプロジェクトで、最小 CPU シェア数と、必要な場合は最大メモリー予約容量 (メモリーキャップ) のリソース予約を設定する必要があります。プロジェクトを使用してこれらの予約の管理を開始する前に、アプリケーションに必要なリソースを把握する必要があります。

注意 – アプリケーションが通常使用するよりも少ない物理メモリーキャップをプロジェクトに設定しないでください。物理メモリーキャップが少ない場合、アプリケーションで仮想メモリーを使用する必要があるので、ページングとスワッピングが多くなり、アプリケーションのパフォーマンスが低下し、大幅な遅延が発生する可能性があります。

プロジェクトを使用してシステムリソースの管理を開始する前に、サーバー統合計画が仕上がっている必要があります。¥また、統合計画に含めるアプリケーションのリソース消費傾向を把握することも重要です。計画を本稼働環境に実装する前に、テスト環境でアプリケーションのリソース消費傾向を 1 か月以上、観察することが理想的です。CPU とメモリーの消費傾向を把握したら、通常の必要なメモリーに数パーセントポイントを上乗せします。

プロジェクトに必要な CPU シェア数を予約するときは、CPU 数を整数で割り当てます。たとえば、25、1、および 37 は有効な値です。「シェア」とは、システムの CPU リソースのうち、プロジェクトに割り当てる分です。プロジェクトに割り当てる CPU シェア数を他のプロジェクトよりも多くすると、そのプロジェクトが公平配分スケジューラから受け取る CPU リソースも多くなります。

CPU シェアは、CPU リソースの比率ではありません。シェアは、他の作業負荷との比較に基づいた作業負荷の相対的な重要性を定義します。たとえば、営業プロジェクトが、マーケティングプロジェクトの 2 倍重要である場合は、営業プロジェクトに、マーケティングプロジェクトの 2 倍のシェア数を割り当てます。割り当てるシェア数は重要ではありません。営業プロジェクトに 2 シェア、マーケティングプロジェクトに 1 シェア割り当てることは、営業プロジェクトに 18 シェア、マーケティングプロジェクトに 9 シェア割り当てることと同じです。どちらの場合も、営業プロジェクトに、マーケティングプロジェクトの 2 倍の CPU 数を使用する権利が与えられます。

CPU シェアは、次の 2 種類に分類できます。

- CPU シェア
- (Solaris 10 のみ) プロジェクトの CPU シェア (特定のゾーン)

プールまたはプロジェクトの作成時の CPU シェア数の割り当て

Solaris 8 OS のホストでは、リソースプール pool_default だけが使用可能です。pool_default の CPU シェア数は 100 です。

Solaris 9 と Solaris 10 OS のホストでは、新規リソースプールを作成すると、プールの CPU シェア数が設定されます。Solaris Container Manager でデフォルト値が入力されますが、任意の整数を入力できます。たとえば、リソースプールに使用可能な CPU 当たり 100 CPU シェアという公式を使用できます。この場合、1 CPU のプールには、100 CPU シェアを割り当てます。

このプールに、Project X、Project Y、および Project Z の 3 つのプロジェクトがあるとします。もっとも重要なプロジェクト Project X に 50 CPU シェア、次のプロジェクト Project Y に 10 シェア、そして次のプロジェクト Project Z に 40 シェア割り当てます。

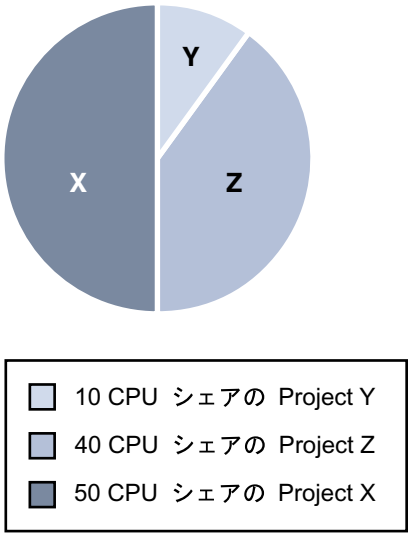


図 **3–8** プロジェクトの CPU シェア数

新規プロジェクトウィザードを使用してプロジェクトを作成するときに、プロジェクトに CPU シェア数を割り当てます。新規プロジェクトウィザードでは、プールの予約されていない CPU シェア数が表示されるので、使用可能な CPU シェア数がわかり、適切な数をプロジェクトに割り当てることができます。

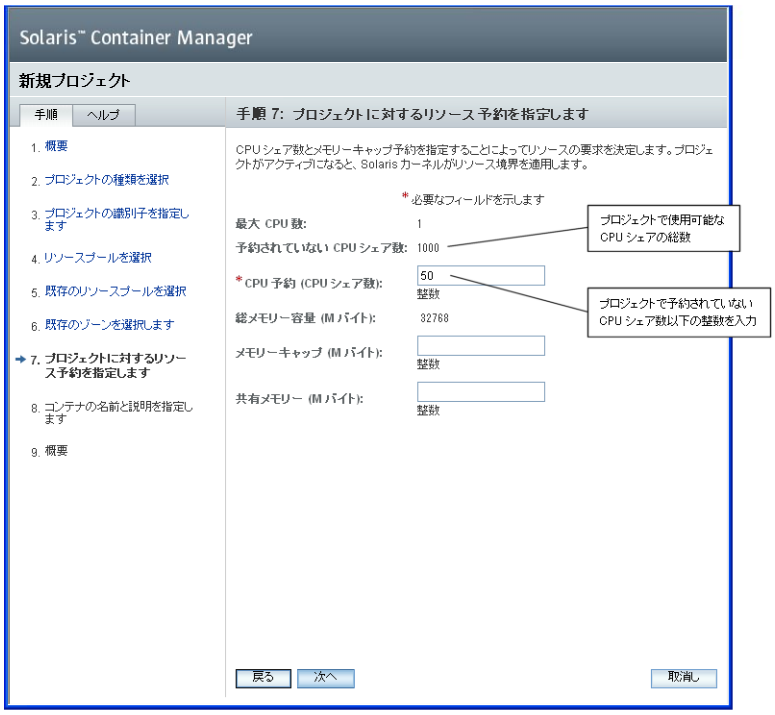


図 **3–9** CPU シェア数

(Solaris 10 のみ) ゾーン作成時の CPU シェア数の割り当て

ホストで Solaris 10 オペレーティングシステムを使用している場合は、ゾーンを作成し、ゾーン全体に CPU 数を割り当て、ゾーン内のプロジェクトにプロジェクトの CPU シェア数を割り当てることができます。これらのシェアは似ています。

CPU シェア数とプロジェクトの CPU シェア数は、新規ゾーンウィザードを使用してゾーンを作成するときに割り当てます。新規ゾーンウィザードの手順 4 で、リソースプールを選択します。すると、プールの総 CPU シェア数と使用可能な CPU シェアの総数が表示されます。

リソースプールからこのゾーンに割り当てる CPU シェア数を入力します。この整数は、プールの使用可能な CPU シェアの総数以下である必要があります。

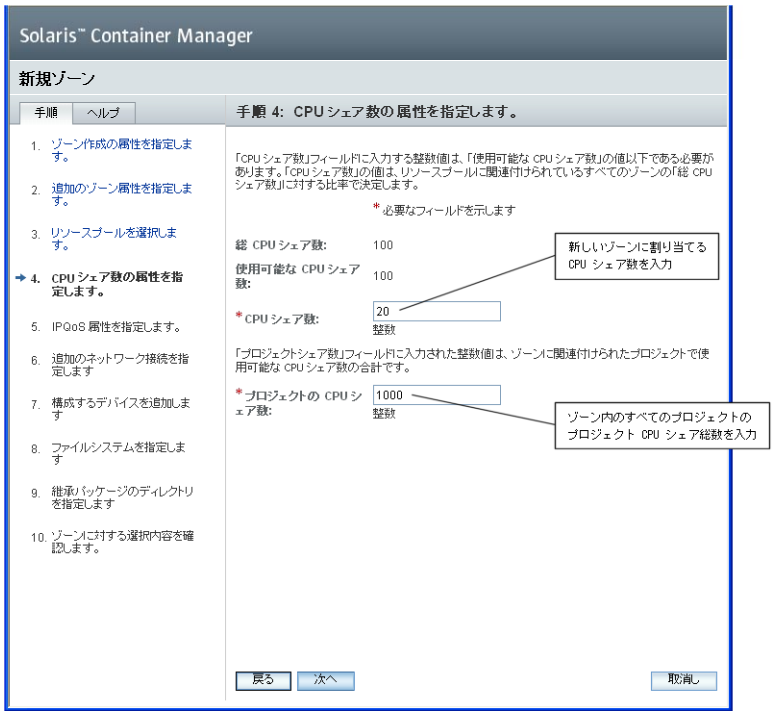


図 **3–10** ゾーンのシェア数

プールの使用可能な CPU シェアの総数が 100 の場合、100 シェアのすべてまたは一部をこのゾーンに割り当てることができます。この例では、20 CPU シェアをリソースプールからゾーンに割り当てています。

ゾーン作成時のプロジェクトの CPU シェア数の割り当て

新規ゾーンウィザードの手順 4 で、プロジェクトの CPU シェア数を入力できます。このフィールドでは、ゾーン内のプロジェクトに割り当てる CPU シェア数を指定します。この値を作成すると、ゾーンのプロジェクトの CPU シェア数が設定されます。整数を入力できます。入力する整数によって、精度が決まります。

たとえば、ゾーン A のプロジェクトの CPU シェア数を 1000 にするとします。物理レベルでは、プロジェクトの CPU シェア数 1000 は、リソースプールから継承された 20 CPU シェアを 1000 シェアに分割したものです。プロジェクトの CPU シェア数 1 つと CPU 数の関係を示す公式を次に示します。

プロジェクトの CPU シェア数 1 = 20 (ゾーンに割り当てられている CPU シェア数)/1000 (プロジェクトの CPU シェア数) = 0.02 CPU シェア

ゾーン A にプロジェクト 1 を作成すると、プロジェクト 1 では、リソースプールから直接ではなく、ゾーンからシェアが取得されます。ゾーン A でプロジェクト 1 に 300 シェアを割り当てた場合、プロジェクトの CPU シェア 300 個 (300/1000 x 20/100 = 0.06 CPU シェア) が割り当てられます。

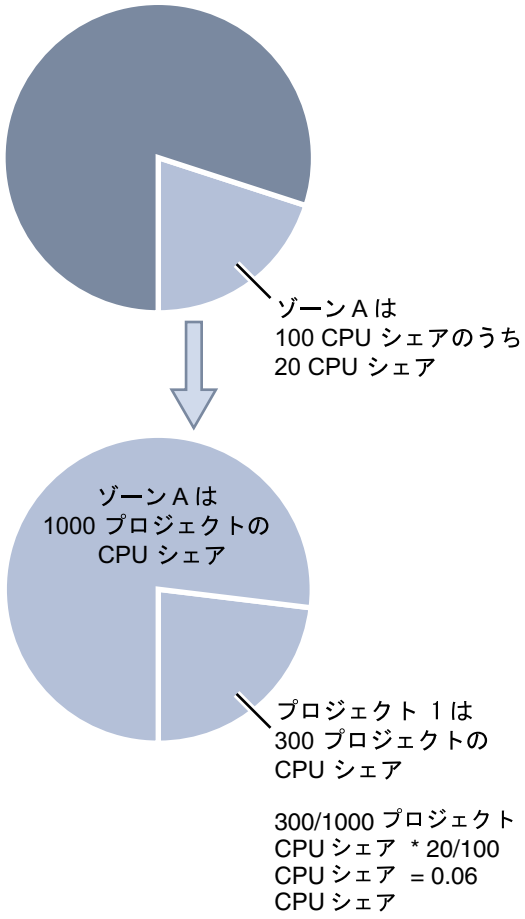


図 **3–11** ゾーンの CPU シェア数

プロジェクトの CPU シェア数は、新規プロジェクトウィザードを起動したときにプロジェクトに割り当てます。新規プロジェクトウィザードの手順 7「プロジェクトに対するリソース予約を指定します」で、「CPU 予約数 (CPU シェア数)」フィールドにプロジェクトの CPU シェア数を入力します。この操作は、Solaris 10 のホストでプロジェクトを作成する場合にのみ可能です。

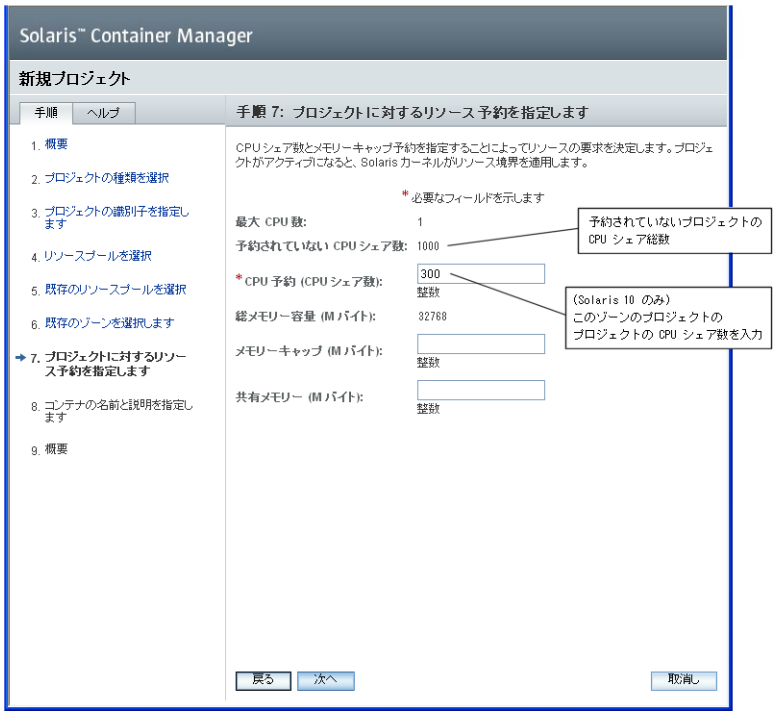


図 **3–12** プロジェクトの CPU シェア数

注 – Solaris 8 または Solaris 9 のホストにプロジェクトを作成するときは、「予約されていない CPU シェア数」フィールドに CPU シェア数 (プロジェクトの CPU シェア数ではない) を入力します。

注意 – コマンド行で zonecfg コマンドを使用して CPU シェア数を手動で変更しないでください。手動で変更すると、Solaris Container Manager の計算が妨害されます。

大域ゾーンとそのプロジェクト

大域ゾーンは、1 つのリソースプールにバインドされていない唯一のゾーンです。大域ゾーンは、任意のプールから CPU リソースを取得できます。ホスト上のすべてのリソースプールには隠れた大域ゾーンがあるので、大域ゾーン内のプロジェクトは、ホストのすべてのリソースプールから CPU リソースを取得できます。

たとえば、リソースプール Pool_default には CPU が 4 つあり、zone_1 と zone_2 が配備されているとします。Pool_default の CPU シェア数は 10 です。zone_1 の CPU シェア数は 5、zone_2 の CPU シェア数は 4、大域ゾーンの CPU シェア数は 1 です。

別のリソースプール Pool_1 には CPU が 2つ、CPU シェアが 10 個あります。Pool_1 には、ゾーン zone_3 だけが配備されています。zone_3 の CPU シェア数は 9 です。大域ゾーンの CPU シェア数は 1 です。

大域ゾーン内のプロジェクトでは、配備されているプールの 1 CPU シェアから CPU リソースが取得されます。

Solaris Container Manager では、大域ゾーン内のプロジェクトは pool_default に配備する必要があります。

公平配分スケジューラ (FSS)

Container Manager では、公平配分スケジューラ (FSS) を使用して、設定した最小 CPU シェア数が確保されます。公平配分スケジューラは、デフォルトのスケジューラです。公平配分スケジューラでは、プロジェクトのシェア数を、有効なプロジェクトのシェアの総数で割って、プロジェクトに割り当てる CPU の割合が計算されます。有効なプロジェクトは、CPU を使用するプロセスが 1 つ以上あるプロジェクトです。アイドル状態のプロジェクト (有効なプロセスがないプロジェクト) のシェア数は、計算に入れられません。

たとえば、営業、マーケティング、およびデータベースの 3 つのプロジェクトにそれぞれ 2 つ、1 つ、および 4 つのシェアが割り当てられるとします。すべてのプロジェクトが有効です。リソースプールの CPU リソースは、営業プロジェクトに 2/7、マーケティングプロジェクトに 1/7、データベースプロジェクトに 4/7 が配分されます。営業プロジェクトがアイドル状態の場合は、マーケティングプロジェクトに 1/5、データベースプロジェクトに 4/5 が配分されます。

公平配分スケジューラでは、CPU の競合がある場合にのみ CPU の使用が制限されます。システムで唯一有効なプロジェクトであるプロジェクトは、シェア数にかかわらず、CPU を 100 パーセント使用できます。CPU サイクルは無駄になりません。あるプロジェクトにおいて、実行する処理がなく、使用する権利があるすべての CPU が使用されていない場合、残りの CPU リソースはほかの有効なプロセスの間で配分されます。プロジェクトの CPU シェア数が定義されていない場合は、1 つのシェアが割り当てられます。シェアが 0 個のプロジェクト内のプロセスは、実行の優先順位が最低になります。このようなプロセスが実行されるのは、シェア数がゼロ以外のプロジェクトが CPU リソースを使用していないときだけです。

タイムシェアスケジューラ (TS)

タイムシェアスケジューラでは、優先順位に基づいて CPU 時間が割り当てられ、使用可能な CPU がすべてのプロセスに比較的均等に配分されます。TS は管理の必要がないので、簡単に使用できます。ただし、TS では、特定のアプリケーションのパフォーマンスは保証されません。TS は、CPU の割り当てが必要ない場合に使用します。

たとえば、2 つのプロジェクトが *FSS* リソースプールに割り当てられ、それぞれに 2 つのシェアがある場合、これらのプロジェクト内で実行されているプロセス数は重要ではありません。1 つのプロジェクトは、使用可能な CPU の 50 パーセントだけを使用できます。したがって、1 つのプロセスが営業プロジェクト内で実行されていて、99 個のプロセスがマーケティングプロジェクト内で実行されている場合、営業プロジェクト内の 1 つのプロセスが CPU の 50 パーセントを使用できます。マーケティングプロジェクト内の 99 個のプロセスは、使用可能な CPU リソースの 50 パーセントを共有する必要があります。

TS のリソースプールでは、CPU がプロセスごとに割り当てられます。営業プロジェクトの 1 つのプロセスは、CPU の 1 パーセントだけを使用でき、マーケティングプロジェクトの 99 個のプロセスが、使用可能な CPU リソースの 99 パーセントを使用できます。

公平配分スケジューラとタイムシェアスケジューラについては、『Solaris のシステム管理 (ネットワークサービス)』を参照してください。

Container Manager の使用によるアプリケーションのリソース消費の管理

次の手順で、テスト環境で Container Manager をツールとして使用して、アプリケーションのリソース消費の傾向を把握できます。

1. Container Manager ソフトウェアと必要なソフトウェアをインストールおよび設定します。

   詳細は、第 2 章を参照してください。

2. 監視するすべてのエージェントマシンに Performance Reporting Manager をインストールします。

   詳細は、第 2 章および『Sun Management Center 3.6 Performance Reporting Manager User's Guide』を参照してください。

3. 傾向を把握するアプリケーション用に、有効なアプリケーションベースのコンテナを作成します。新規プロジェクトウィザードで、最小 CPU 予約数だけを設定します。メモリーキャップは設定しません。

   詳細は、80 ページの「アプリケーションベースのプロジェクトの作成」および82 ページの「アプリケーションベースのプロジェクトを作成する」を参照してください。

4. 日別、週別、またはリアルタイムのグラフで、使用されているリソースを数週間、監視します。1 つのホストで実行中のコンテナの CPU とメモリーリソースの 2 つのグラフを確認できます。また、「プロセス」表でアプリケーション内で実行中のプロセスを監視することもできます。

   詳細は、128 ページの「有効なプロジェクトのリソース使用状況レポートを要求する」および91 ページの「プロジェクトのプロセスの表示」を参照してください。

5. アプリケーションに必要な物理メモリーの最大容量を把握したら、コンテナのプロパティを変更してメモリーキャップを設定します。メモリーキャップは、アプリケーションで使用される最大メモリー容量以上にします。

詳細は、96 ページの「プロパティシートを使用してプロジェクトを変更する」を参照してください。

6. 使用メモリーが、設定したメモリーキャップを超え始めたときに通知を受け取るようにアラームを設定します。必要な場合は、プロパティシートでメモリーキャップを調整します。

   詳細は、120 ページの「アラームのしきい値を設定する」および96 ページの「プロパティシートを使用してプロジェクトを変更する」を参照してください。

Container Manager を使用してリソース使用状況の傾向を把握したら、コンテナを使用して本稼働環境でサーバーを統合できます。

サーバーの統合の計画および実行については、David Hornby、Ken Pepple 共著『Consolidation in the Data Center』(Sun Blueprints) を参照してください。ORACLE データベースを使用しているシステムのサーバー統合については、Sun のホワイトペーパー『Consolidating Oracle RDBMS Instances Using Solaris Resource Manager Software』を参照してください。

第 4 章

プロジェクトの管理

この章では、プロジェクトを作成、使用、および管理する手順を示します。

この章では、以下の内容について説明します。

- 73 ページの「プロジェクトの作成」
- 85 ページの「プロジェクト内でのプロセスの移動または起動」
- 88 ページの「プロジェクトの有効化と無効化」
- 91 ページの「プロジェクトのプロセスの表示」
- 93 ページの「コンテナおよびプロジェクトの変更」
- 123 ページの「レポート」
- 101 ページの「プロジェクトの削除」

注 – プロジェクトを管理 (作成、有効化、無効化、変更、削除) するには、プロジェクト管理者権限が必要です。プロジェクト管理者は、Solaris Container Manager ソフトウェアを設定する際に指定します。

プロジェクトの作成

ソフトウェアのインストールと設定後に作成されるデフォルトコンテナに加えて、カスタムプロジェクトを作成できます。両方のタイプのコンテナを組み合わせて使用すると、サーバー統合計画の実行に役立ちます。

カスタムプロジェクトを作成するには、新規プロジェクトウィザードを使用します。コンテナだけを作成し、コンテナ表示ウィンドウに保存できます。また、ウィザードの手順をすべて完了してプロジェクトを作成することもできます。どちらの場合も同じウィザードを使用します。

コンテナだけを作成した場合、名前がコンテナ表示に保存されます。あとでコンテナを使用して 1 つまたは複数のプロジェクトを作成できます。詳細は、88 ページの「プロジェクトの有効化と無効化」を参照してください。

プロジェクトを作成する場合も、その一環としてコンテナを作成します。プロジェクトの作成が完了したら、コンテナがコンテナ表示のナビゲーションウィンドウに保存されます。この同じ定義を使用して、複数のホストに関連付けられている追加プロジェクトを作成できます。これらのプロジェクトの定義は、名前とプロジェクトの種類を含めて、各ホストで同じです。プロジェクトのリソース予約はホストごとに変えるか、すべて同じにできます。状況に応じてリソースのニーズを満たせるように、このように柔軟な設定が可能になっています。詳細は、47 ページの「コンテナのプロパティー」を参照してください。

新規プロジェクトウィザードに従って、プロジェクトを作成できます。ウィザードに従ってプロジェクトを作成するには、次の情報が必要です。

- プロジェクトの名前と説明。
- プロジェクトの種類。詳細は、表 3–2 を参照してください。
- プロジェクトを関連付けるホストの名前。
- プロジェクトをバインドするリソースプールの名前。
- プロジェクトに関連付けるリソース境界 (最小 CPU 予約数とメモリキャップ容量)。

このウィザードは、GUI の 3 つの異なる場所で「新規プロジェクト」ボタンをクリックして起動できます。GUI のどこからウィザードを起動するかによって、上記の情報をすべて指定する必要がない場合があります。この場合、一部の情報は自動的に入力されます。

▼ 新規プロジェクトウィザードを起動する

新規プロジェクトウィザードは、GUI の 3 つの場所から起動できます。ウィザードを起動した場所によっては、一部の情報が自動的に入力されるので、すべてのパネルに入力する必要がない場合があります。

新規プロジェクトウィザードの例については、80 ページの「アプリケーションベースのプロジェクトの作成」を参照してください。

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで、作成するコンテナの関係を指定します。
   - 特定のホストを自動的にコンテナに関連付けるには、次の操作を行います。
     a. ホスト表示のナビゲーションウィンドウでホスト名を選択します。
        必要な場合は、ホストグループ名をクリックしてリストを展開します。
     b. 右側の区画で「プロジェクト」タブを選択します。
        「プロジェクト」表が表示されます。

     この方法では、コンテナの作成処理中にホストを選択する必要があります。

- プロジェクトを特定のリソースプールに自動的にバインドするには、次の操作を行います。

  a. ホスト表示のナビゲーションウィンドウでリソースプール名を選択します。

     必要な場合は、ホスト名の横のボタンをクリックしてリストを展開します。ホストに割り当てられているリソースプールが表示されます。

  b. 右側の区画で「プロジェクト」タブを選択します。

     「プロジェクト」表が表示されます。

  コンテナの作成処理中にリソースプールを割り当てる必要はありません。

3. ホストプロジェクトまたはホストリソースプールの表で「新規プロジェクト」ボタンをクリックします。

注 – どちらの方法でも、右側の区画に表示される表に「新規プロジェクト」ボタンがあります。

新規プロジェクトウィザードが表示されます。最初に「概要」パネルが表示されます。

ユーザーベースまたはグループベースのプロジェクトの作成

注 – Solaris 8 では、ユーザーベースのコンテナだけを使用できます。

コンテナで、UNIX ユーザー名または UNIX グループ名で識別されるプロセスを管理するには、ユーザーベースまたはグループベースのコンテナを作成します。作成処理中に選択するプロジェクトの種類によって、完成したコンテナがユーザーベースであるか、グループベースであるかが決まります。

▼ ユーザーベースまたはグループベースのプロジェクトを作成する

手順 **1.** 74 ページの「新規プロジェクトウィザードを起動する」に従って新規プロジェクトウィザードを起動します。

「概要」パネルが表示されます。

2. プロジェクトの種類として「ユーザー」または「グループ」を選択します。

プロジェクトの種類が「ユーザー」のコンテナでは、同じ UNIX ユーザー名のプロセスが追跡されます。

プロジェクトの種類が「グループ」のコンテナでは、同じ UNIX グループ名のプロセスが追跡されます。

注 – Solaris 8 OS では、ユーザーベースのコンテナだけを使用できます。

---

3. 次の要件を満たすプロジェクトの種類の識別子を指定します。

(Solaris 8)

- ユーザーベースのプロジェクト - 最初のフィールドに有効な UNIX ユーザー名を入力します。プロジェクトに参加できるユーザーの UNIX ユーザー名を 2 番目のフィールドに追加できます。プロジェクトに参加できるグループの UNIX グループ名を 3 番目のフィールドに追加できます。複数の名前を入力する場合は、コンマで区切ります。

  別のユーザーベースのプロジェクトまたはデフォルトプロジェクトで使用されているユーザー名を最初のフィールドに入力しないでください。
- グループベースのプロジェクト - 最初のフィールドに有効な UNIX グループ名を入力します。あとでプロジェクトに加わることができるグループの UNIX グループ名を 2 番目のフィールドに追加できます。あとでプロジェクトに加わることができるユーザーの UNIX ユーザー名を 3 番目のフィールドに追加できます。複数の名前を入力する場合は、コンマで区切ります。

  別のグループベースのプロジェクトまたはデフォルトプロジェクトで使用されているグループ名を最初のフィールドに入力しないでください。

注 – Solaris 9 と Solaris 10 では、有効なユーザー名とグループ名は必須ではありません。ただし、追加のユーザー名とグループ名がシステムに存在する必要があります。

このプロジェクトの種類については、表 3–2 を参照してください。

4. ホスト名だけを選択して、プールを選択せずに新規プロジェクトウィザードを起動した場合は、プロジェクトのリソース要件を満たすリソースプールを割り当てます。

   プロジェクト内で開始された新規プロセスは、対応するリソースプールにバインドされます。プロジェクトが有効になったら、プロジェクトで保持されている新規プロセスがそのリソースプールにバインドされます。

   - 新規リソースプールを割り当てるには、次の操作を行います。

     a. 「新規リソースプールを作成」を選択します。

     b. リソースプールの名前を指定します。

        名前には英数字を使用し、空白文字を含めることはできません。ハイフン (-)、下線 (_)、およびピリオド (.) の各文字は使用できます。

     c. CPU 数を割り当てます。

        CPU 数は、ホスト上で使用可能な CPU 数以下、1 以上の整数である必要があります。ホスト上の CPU の総数と、現在、使用可能な数が表示されます。

   - 既存のプールを割り当てるには、次の操作を行います。

     a. 「既存のリソースプールを使用」を選択します。

        使用可能なリソースプールのリストが表示されます。

     b. リスト内のプール名の横のラジオボタンを選択します。

        各リソースプールに割り当てられている CPU の総数と、各プール内で予約されていない CPU 数が表示されます。コンテナが、選択したリソースプールに割り当てられます。

5. **(Solaris 10 のみ)** 既存のゾーンを選択します。

   使用可能なゾーンのボタンをクリックし、プロジェクトをゾーンに関連付けます。

6. プロジェクトのリソース予約を指定します。

   ホストの最大 CPU 数、予約されていない CPU シェア数、および使用可能なメモリーリソースを指定します。

   CPU 予約数は必須であり、整数で指定する必要があります。CPU 予約数がゼロのコンテナは、同じホストに関連付けられているほかのコンテナ内で実行中のプロセスがない場合にのみ CPU リソースを使用できます。

   メモリーキャップは省略可能で、M バイト単位で指定します。

   共有メモリーの値は、このプロジェクト内で実行されるプロセスで使用できる共有メモリーの総容量です。共有メモリーも省略可能です。共有メモリーのデフォルト値は、物理メモリーの 4 分の 1 です。

7. コンテナの名前を指定します。

   名前は、一意であり、32 文字以内である必要があります。ナビゲーションウィンドウ、各一覧表、およびリソース使用状況レポートでこの名前でコンテナが識別されます。重複する名前を入力すると、コンテナの作成に失敗します。

作成処理の完了後は、コンテナ名を変更できません。

8. **(省略可能)** コンテナの説明を入力します。

9. 「概要」パネルで設定を確認します。

10. 「完了」をクリックします。

    設定が保存され、プロジェクトが有効になります。Solaris カーネルがコンテナのリソース予約を開始します。

アプリケーションベースのプロジェクトの作成

注 – Solaris 8 OS では、ユーザーベースのコンテナだけを使用できます。

アプリケーションベースのプロジェクトを使用して、特定のソフトウェアアプリケーションで実行されるプロセスを管理します。プロセスが自動的に移動するか、プロセスを手動で移動できるアプリケーションベースのプロジェクトを作成できます。

アプリケーションに固有の一致式を指定できる場合は、この式をプロジェクトに追加できます。この場合、プロセスの実行に使用される UNIX ユーザー ID または UNIX グループ ID も指定する必要があります。あとでプロジェクトに加わることができるユーザーまたはグループも追加できます。プロジェクトにプロセスを自動的に移動するには、ウィザードの対応するパネルが表示されたときに、必要なプロジェクト識別子をすべて指定する必要があります。識別子を指定すると、この定義に基づくすべてのプロジェクトで、一致するプロセスがすべて自動的に移動します。

アプリケーションに固有の識別子が作成されない場合は、プロセスを手動で移動するか、プロジェクト内でアプリケーションを起動できます。プロセスを手動で移動する場合は、プロセスの実行に使用される UNIX ユーザー ID または UNIX グループ ID だけでプロジェクトを作成します。あとでプロジェクトに加わることができるユーザーまたはグループも追加できます。そのあと、`newtask -p` コマンドでプロセスを移動します。詳細は、85 ページの「プロジェクト内でのプロセスの移動または起動」を参照してください。

▼ アプリケーションの一致式を決定する

管理するアプリケーションに対応するプロセスを識別する適切な一致式を決定するには、次の手順に従います。プロセスをコンテナに自動的に移動するには、新規コンテナウィザードでこの式を指定する必要があります。

手順

1. 端末ウィンドウから、アプリケーションベースのコンテナで管理するアプリケーションを起動します。
2. 実行中のすべてのプロセスの一覧を表示するには、端末ウィンドウで次のように入力します。

   ```
   % ps -cafe
   ```

3. **CMD** 列で対応する実行可能ファイルの名前を探します。
   アプリケーションのプロセスを一意に識別する式を選択します。

例 4–1 Mozilla の一致式の決定

Mozilla を探すために入力した `ps - cafe` コマンドの出力例を次に示します。

```
% ps -cafe
     UID   PID  PPID  CLS PRI    STIME TTY      TIME CMD
    ...
username  8044  7435   IA  50 19:47:09 pts/11   0:00 /bin/ksh -p /usr/sfw/lib/mozilla/mozilla
```

この例では、固有の実行可能ファイルの名前は `mozilla` です。したがって、適切な一致式は `mozilla` になります。

例 4–2 Tomcat サーバーの一致式の決定

アプリケーションの名前がわかっている場合は、`ps -cafe` コマンドと `grep` コマンドを組み合わせて使用して適切な一致式を確認できます。Tomcat サーバーを探すために入力した `ps - cafe | grep tomcat` コマンドの出力例を次に示します。この例では、関係のない情報を省いています。

```
% ps -cafe | grep tomcat
  nobody 27307  /usr/j2se/bin/java -classpath //usr/apache/tomcat/bin/bootstrap.jar:/usr/j2se/l
```

この例では、実行可能ファイルの名前は `java` です。ただし、適切な一致式は `tomcat` です。この場合、一致式は、実行可能ファイルの名前ではなく、引数です。なぜなら、`java` では、Tomcat のプロセスを一意に識別できないからです。

例 4–3 Tomcat サーバーの一致式の確認

次の例では、`pgrep` コマンドを使用して PID を検索する方法を示します。PID は、必要なプロセスを検索するための固有の一致式を指定したことを示します。

```
% pgrep -f tomcat
27307
```

Tomcat サーバーの PID は `27307` です。この数値は、例 4–2 の PID と一致します。したがって、一致式 tomcat が Tomcat サーバーのプロセスと一致することが確認できます。

▼ アプリケーションベースのプロジェクトを作成する

手順 **1.** **74** ページの「新規プロジェクトウィザードを起動する」に従って新規プロジェクトウィザードを起動します。

「概要」パネルが表示されます。

2. コンテナのプロジェクトの種類として「アプリケーション」を選択します。

アプリケーションベースのプロジェクトコンテナでは、アプリケーションに関連付けられているプロセスが追跡されます。このプロジェクトの種類については、表 3–2 を参照してください。

3. プロジェクトの名前を指定します。

名前は、一意であり、32 文字以内である必要があります。ナビゲーションウィンドウ、各一覧表、およびリソース使用状況レポートでこの名前でプロジェクトが識別されます。重複する名前を入力すると、プロジェクトの作成に失敗します。

作成処理の完了後は、コンテナ名を変更できません。

4. アプリケーションのプロセスの実行に使用される **UNIX** ユーザー名または **UNIX** グループ名を指定します。

この UNIX ユーザー名または UNIX グループ名は必須です。名前を指定しなかった場合、名前を指定するまで、対応するプロセスがコンテナ下に移動しませ

ん。複数の名前を入力する場合は、コンマで区切ります。

5. コンテナが有効になったときにアプリケーションのプロセスをプロジェクト下に自動的に移動するか、コマンド行から手動で移動するかを指定します。

   - アプリケーションのプロセスをコマンド行から手動で移動するには、「一致式を使用しない」チェックボックスを選択します。
   - プロジェクトが有効になったときにアプリケーションのプロセスをプロジェクト下に自動的に移動するには、「一致式」フィールドに式を指定します。

     「一致式」フィールドでは、アプリケーションに関連付けられているプロセスを識別するワイルドカードを使用できます。たとえば、Mozilla に関連するプロセスは moz、Tomcat に関連するプロセスは cat と指定できます。

アプリケーションのプロセスを自動的にコンテナに移動するには、一致式を指定する必要があります。一致式では、大小文字が区別されます。適切な一致式を決定するには、81 ページの「アプリケーションの一致式を決定する」を参照してください。

この時点で一致式を指定しなかった場合、式を指定するまで、アプリケーションのプロセスはこのコンテナ下に移動しません。

6. ホスト名から、プールを選択せずに新規プロジェクトウィザードを起動した場合は、プロジェクトのリソース要件を満たすリソースプールを割り当てます。

   プロジェクト内で開始された新規プロセスは、対応するリソースプールにバインドされます。プロジェクトが有効になったら、コンテナで保持されている新規プロセスがそのリソースプールにバインドされます。

   - 新規リソースプールを割り当てるには、次の操作を行います。
     a. 「新規リソースプールを作成」を選択します。
     b. リソースプールの名前を指定します。

        名前には英数字を使用し、空白文字を含めることはできません。ハイフン (-)、下線 (_)、およびピリオド (.) の各文字は使用できます。
     c. CPU 数を割り当てます。

        CPU 数は、ホスト上で使用可能な CPU 数以下、1 以上の整数である必要があります。ホスト上の CPU の総数と、現在、使用可能な数が表示されます。
   - 既存のプールを割り当てるには、次の操作を行います。
     a. 「既存のリソースプールを使用」を選択します。

        使用可能なリソースプールのリストが表示されます。
     b. リスト内のプール名の横のラジオボタンを選択します。

        各リソースプールに割り当てられている CPU の総数と、各プール内で予約されていない CPU 数が表示されます。プロジェクトが、選択したリソースプールにバインドされます。

7. **(Solaris 10 のみ)** 既存のゾーンを選択します。

   使用可能なゾーンのボタンをクリックします。

8. プロジェクトのリソース予約を指定します。

   ホストの最大 CPU 数、予約されていない CPU シェア数、および使用可能なメモリーリソースを指定します。

   CPU 予約数 (CPU シェア数) は必須で、整数で指定する必要があります。CPU 予約数がゼロのプロジェクトは、同じホストに関連付けられているほかのコンテナ内で実行中のプロセスがない場合にのみ CPU リソースを使用できます。

   メモリーキャップは省略可能で、M バイト単位で指定します。

   共有メモリーの値は、このプロジェクト内で実行されるプロセスで使用できる共有メモリーの総容量です。共有メモリーも省略可能です。共有メモリーのデフォルト値は、物理メモリーの 4 分の 1 です。

9. コンテナの名前を指定します。

   名前は、一意であり、32 文字以内である必要があります。ナビゲーションウィンドウ、各一覧表、およびリソース使用状況レポートでこの名前でコンテナが識別されます。重複する名前を入力すると、コンテナの作成に失敗します。作成処理の完了後は、コンテナ名を変更できません。

10. (省略可能) コンテナの説明を入力します。

11. 「概要」パネルで設定を確認します。

12. 「完了」をクリックします。

設定が保存され、コンテナが有効になります。Solaris カーネルがコンテナのリソース予約を開始します。

プロジェクト内でのプロセスの移動または起動

プロジェクトで管理するアプリケーションに一意の実行可能ファイル名がない場合は、プロセスを手動でプロジェクトに移動できます。この方法によって、必要なアプリケーションのプロセスだけを追跡できます。

▼ プロセスをアプリケーションベースのプロジェクトに個別に移動する

この手順は、アプリケーションベースのプロジェクトに一致式を指定せず、アプリケーションのプロセスを個別にプロジェクトに移動する場合に使用します。

手順

1. アプリケーションを管理するアプリケーションベースのプロジェクトを作成します。「一致式を使用しない」チェックボックスを選択します。
   詳しい手順については、82 ページの「アプリケーションベースのプロジェクトを作成する」を参照してください。
2. 次のように入力して、**/etc/project** ファイルでプロジェクト名を確認します。

   ```
   % cat /etc/project
   ```

   このプロジェクト名は手順 5 で必要です。
3. 必要な場合は、端末ウィンドウでアプリケーションを起動します。
4. アプリケーションに対応するプロセスを確認します。
   例については、例 4–1、例 4–2、および例 4–3 を参照してください。
5. 対応するプロセスを移動します。
   a. **Container Manager** の **GUI** で、ナビゲーションウィンドウでホストをクリックします。
   b. 「プロジェクト」タブをクリックします。
      「プロジェクト」表が表示されます。
   c. 「プロジェクト」表でプロジェクト名をクリックします。プロジェクト名には、リンクであることを示す下線が表示されます。
      「プロパティ」タブが表示されます。
   d. 「プロセス」タブをクリックします。
      「プロセス」表が表示されます。
   e. 移動するプロセスを選択します。
   f. 「移動」ボタンをクリックします。
      「プロセスを移動」ダイアログが表示されます。
   g. プロジェクト **(**コンテナ**)** のリストから、プロセスを移動する新規プロジェクト **(**コンテナ**)** を選択します。
   h. 「了解」をクリックします。
6. プロセスをすべて移動するまで手順 **5** を繰り返します。

例 4–4 ps を使用したプロセスのプロジェクトへの移動の確認

ps コマンドと grep コマンドを組み合わせて使用して、プロセスがコンテナに移動したことを確認できます。次の例は、個別に移動したプロセスが「payroll」というコンテナにあることを示します。

```
% ps -ae -o pid,project,comm | grep payroll
17773    payroll ora_reco_AcctEZ
17763    payroll ora_pmon_AcctEZ
17767    payroll ora_lgwr_AcctEZ
```

例 4–5 prstat を使用したプロセスのプロジェクトへの移動の確認

プロジェクト名がわかる場合は、prstat コマンドを使用してプロセスがコンテナに移動したことを確認できます。この例では、プロジェクト名は「payroll」です。

```
% prstat -J payroll
   PID USERNAME  SIZE   RSS STATE  PRI NICE      TIME  CPU PROCESS/NLWP
 17773 admin     216M  215M cpu2     1    0   0:05:08  29% ora_reco_AcctEZ/1
 17763 admin     834M  782M sleep    1    0   0:35:02   0% ora_pmon_AcctEZ/1
 17767 admin     364M  352M run      1    0   0:22:05  23% ora_lgwr_AcctEZ/1
```

▼ プロジェクト内でアプリケーションを起動する

手順

1. アプリケーションを管理するアプリケーションベースのプロジェクトを作成します。「一致式を使用しない」チェックボックスを選択します。

   詳しい手順については、82 ページの「アプリケーションベースのプロジェクトを作成する」を参照してください。

2. **OS** のバージョンによって、次のいずれかの操作を行います。

   - Solaris 8 OS の場合、次のように入力します。

     ```
     % srmuser user_name newtask -p project_name application_name
     ```

     user_name は UNIX ユーザー名、*project_name* は user.*username* の書式で指定します。Solaris 8 OS では、ユーザーベースのコンテナだけを使用できるので、*user_name* と *project_name* は同じです。

   - Solaris 9 または Solaris 10 OS の場合、次のように入力します。

     ```
     % newtask -p project_name application_name
     ```

     project_name は、コンテナに関連付けられているプロジェクト、*application_name* は、アプリケーションを起動するコマンドです。コマンドには、コマンド引数も含めます。

   アプリケーションがコンテナ内で起動します。

例 4–6 Solaris 9 または Solaris 10 OS でのコンテナ内でのアプリケーションの起動

music というコンテナ内で tracks というアプリケーションを起動する例を次に示します。

```
% newtask -p music tracks -z 0 mozart.au
```

-z 0 mozart.au は、アプリケーション tracks のコマンド行引数です。

例 4–7 アプリケーションとプロジェクトの関連付けの確認

アプリケーションが起動したら、次のように入力して、アプリケーションが関連付けられているプロジェクトを確認できます。

```
% ps -ae -o pid,project,comm
```

このコマンドの出力例を次に示します。

```
  PID  PROJECT COMMAND
...
17771   default ora_smon_SunMC
16246   system rquotad
26760   group.staff /bin/csh
16266   music     tracks
17777   default ora_d000_SunMC
17775   default ora_s000_SunMC
17769   default ora_ckpt_SunMC
```

この例では、アプリケーション tracks の PID は 16266、プロジェクトは music、実行可能ファイルは tracks です。これは、例 4–6 で起動したアプリケーションと同じです。

プロジェクトの有効化と無効化

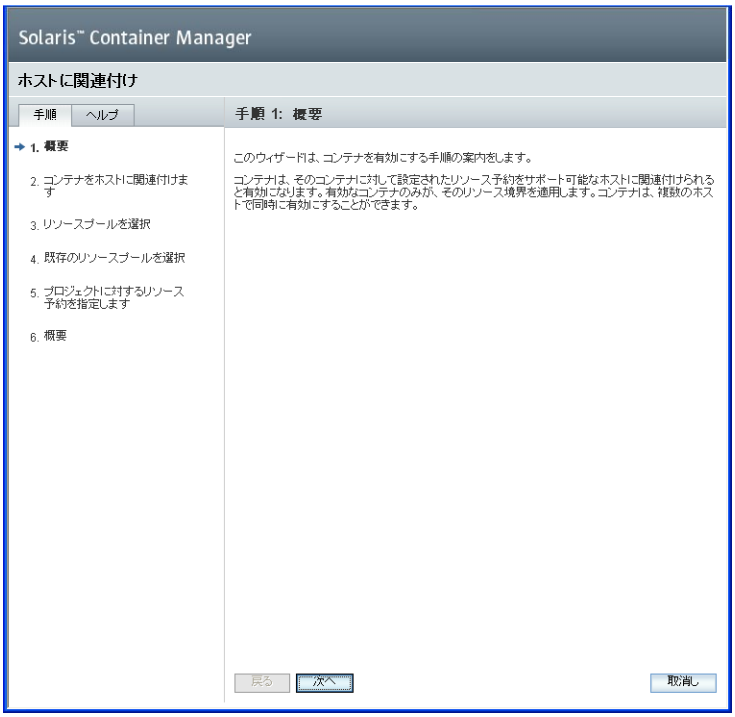
プロジェクトが定義済みまたは無効な状態のときは、プロジェクトのリソース境界は適用されません。リソース境界を適用するには、プロジェクトを有効にする必要があります。また、この制限を適用しない場合は、プロジェクトを無効にする必要があります。有効なプロジェクトを無効にしても、設定したリソース境界は失われません。詳細は、49 ページの「プロジェクトの状態」を参照してください。

ホストに関連付けウィザードで、既存のコンテナを使用して有効なプロジェクトを新規に作成できます。無効なプロジェクトを有効にするか、有効なプロジェクトを無効にするには、ボタンを使用します。

▼ プロジェクトを有効にする

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. コンテナ表示で、コンテナの名前を選択します。
   コンテナがグループに含まれる場合は、ナビゲーションウィンドウでそのグループを選択して右側の区画にコンテナを表示します。
3. 右側の区画で「ホスト」タブを選択します。
   「このコンテナに関連付けられたホスト」表が表示されます。選択したコンテナに現在関連付けられているすべてのホストが表に表示されます。
4. 「ホストに関連付け」ボタンをクリックします。
   ホストに関連付けウィザードが表示されます。



5. ウィザードの指示に従い、ホスト名を入力するか、該当するホストを検索します。

▼ 無効なプロジェクトを有効にする

手順 1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. ホスト表示で、コンテナが関連付けられているホストを選択します。

3. 右側のパネルで「プロジェクト」を選択します。
   このホストに関連付けられているすべてのプロジェクトを示す表が表示されます。

4. 「有効にする」ボタンを使用可能にするには、有効にするプロジェクトのチェックボックスを選択します。

5. **(**省略可能**)** プロジェクトのプロパティを表示し、変更します。

   a. 表の「プロジェクト名」列でプロジェクト名をクリックします。
      「プロパティ」タブが表示されます。

   b. **(**省略可能**)** プロジェクトのリソース予約を変更し、「保存」をクリックします。

   詳細は、93 ページの「コンテナおよびプロジェクトの変更」を参照してください。

6. 「有効にする」ボタンをクリックします。
   プロジェクトが有効になり、カーネルによってリソース境界が適用されます。

▼ 有効なプロジェクトを無効にする

手順 1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. ホスト表示で、プロジェクトが関連付けられているホストを選択します。

3. 右側のパネルで「プロジェクト」を選択します。
   このホストに関連付けられているすべてのプロジェクトを示す表が表示されます。

4. 「無効にする」ボタンを使用可能にするには、無効にするプロジェクトのチェックボックスを選択します。

5. **(**省略可能**)** プロジェクトのプロパティを表示し、変更します。

   a. 表の「プロジェクト名」列でプロジェクト名をクリックします。
      「プロパティ」タブが表示されます。

   b. **(**省略可能**)** プロジェクトのリソース予約を変更し、「保存」をクリックします。

詳細は、93 ページの「コンテナおよびプロジェクトの変更」を参照してください。

6. 「無効にする」ボタンをクリックします。
   プロジェクトが無効になり、カーネルによってリソース境界が適用されなくなります。

プロジェクトのプロセスの表示

有効なプロジェクト内で実行中のプロセスに関する情報は、ホスト表示またはコンテナ表示の表で確認できます。どちらのビューでも、同じ「プロセス」表と、同じ情報が表示されます。

プロセスは、1 行に 1 つずつ表示され、次の情報が表示されます。

PID	プロセス ID
ユーザー名	プロセスの所有者 (UNIX ユーザー名またはログイン名)
サイズ	プロセスの仮想メモリーサイズ (M バイト単位)
RSS	プロセスの常駐セットのサイズ (M バイト単位)
状態	プロセスの状態。次の値があります。

- cpuN – プロセスは CPU *N* で実行されています。*N* は整数です。

	■ sleep – プロセスはスリープまたは待機しています。 ■ run – プロセスは実行中です。 ■ zombie – プロセスは終了しています。 ■ stop – プロセスは停止しています。
PRI	プロセスの優先順位。数値が大きいほど、プロセスの優先順位が高くなります。
NICE	優先順位の計算に使用される nice 値
TIME	プロセスの累積実行時間
CPU	現在のプロセスで使用されている時間の割合
プロセス/NLWP	プロセスの名前、つまり実行されているファイルの名前。プロセス内の軽量プロセス (LWPS) 数。

▼ プロジェクト内で実行中のプロセスを「ホスト」表示で表示する

プロジェクトが関連付けられているホストの名前がわかる場合は、この手順に従います。

手順

1. **Container Manager** の **GUI** が開いていない場合は、51 ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで左側のタブをクリックしてホスト表示を選択します。
3. ナビゲーションウィンドウで、プロジェクトが関連付けられているホストを選択します。
4. 右側の区画で「プロジェクト」タブを選択します。
   「プロジェクト」表が表示され、ホストに関連付けられているプロジェクトがすべて表示されます。表には有効なプロジェクトと無効なプロジェクトが含まれます。プロセスの情報を表示するには、有効なプロジェクトを選択します。
5. プロジェクト名をクリックしてプロジェクトを選択します。
   選択したホストのプロジェクトインスタンスのプロパティページが表示されます。
6. 「プロセス」タブを選択します。
   プロジェクト内で実行中のプロセスが「プロセス」表に表示されます。プロジェクトの名前と、そのプロジェクトが関連付けられているホストが、表の上に表示されます。

   プロセスが表示されない場合は、無効なコンテナが選択されている可能性があります。

▼ プロジェクト内で実行中のプロセスを「コンテナ」表示で表示する

プロジェクトの名前がわかっていて、プロジェクトが関連付けられているホストのリストから選択する場合は、この手順に従います。

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで右側のタブをクリックしてコンテナ表示を選択します。
3. ナビゲーションウィンドウで、表示するコンテナを選択します。

   必要な場合は、三角形の展開ボタンをクリックしてコンテナを展開します。プロジェクト名が表示されたら、表示するプロジェクトの名前をクリックします。

   コンテナが関連付けられているホストをすべて含む表が右側の区画に表示されます。
4. 表でプロジェクト名をクリックします。

   選択したホスト上のコンテナインスタンスのプロパティページが表示されます。
5. 「プロセス」タブを選択します。

   プロジェクトの名前と、そのプロジェクトが関連付けられているホストが、表のタイトルに表示されます。プロジェクト内で実行中のプロセスが「プロセス」表に表示されます。

   表にプロセスが表示されない場合は、無効なプロジェクトが選択されている可能性があります。

コンテナおよびプロジェクトの変更

コンテナ、あるいは有効か無効なプロジェクトは、2 つのプロパティシートで変更できます。各プロパティシートの違いを次の表に示します。

表 **4–1** プロパティシートの詳細

プロパティシート	用途	タブ
コンテナ	説明、プロジェクトの種類、プロジェクトの識別子 (ユーザー、グループ)、一致式の変更	コンテナ表示でコンテナを選択後の「プロパティ」タブ
プロジェクトインスタンス (有効または無効なプロジェクト)	リソースプールの関連付け、CPU 予約数、メモリーキャップの変更	ホスト表示またはコンテナ表示の「プロパティ」タブ

プロジェクトインスタンスには、それぞれ関連付けられているコンテナがあります。コンテナに対する変更は、その定義を使用するすべてのプロジェクトインスタンスに適用されます。たとえば、コンテナでプロジェクトの種類を変更すると、同じコンテナを使用するすべてのプロジェクトインスタンスのプロジェクトの種類が変更されます。したがって、両方のプロパティシートを使用してすべての変更を行うことができます。

各プロジェクトインスタンスには、固有のリソースプールの関連付けまたはリソース予約だけを変更するためのプロパティシートもあります。このプロパティシートを使用するときは、一度に 1 プロジェクトずつ変更できます。たとえば、現在の最小 CPU 予約数または現在のメモリーキャップを増加できます。変更内容は、値を保存すると有効になります。無効なプロジェクトのリソース境界への変更は、プロジェクトを有効にするまで有効になりません。

sales01 プロジェクトインスタンスのプロパティシートの例を次に示します。

図 **4–1** 例: リソース予約とリソースプールを変更するプロパティシート

複数のホストで有効な複数のプロジェクトのリソースを変更するには、リソース変更ジョブ機能を使用します。詳細は、97 ページの「リソース変更ジョブによるプロジェクトの変更」を参照してください。

コンテナ表示から、コンテナを変更できる別のプロパティシートを使用できます。一度に 1 コンテナ ずつを変更できます。リソース変更ジョブ機能を使用して複数のコンテナを変更することはできません。

sales01 プロジェクトインスタンスのプロパティシートの例を次に示します。

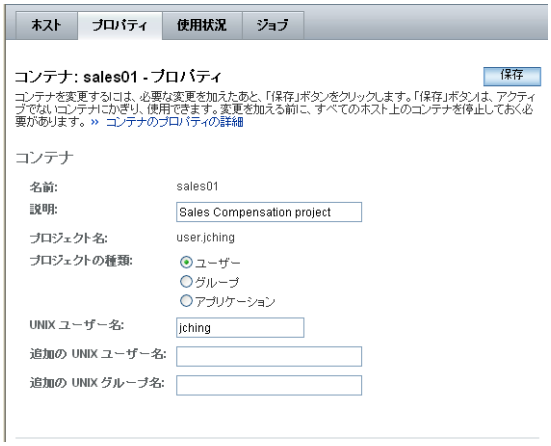


図 **4–2** 例: コンテナを変更するプロパティシート

デフォルトプロジェクトのプロパティは変更できません。したがって、デフォルトプロジェクトを選択した場合は、どちらのプロパティシートも使用できません。

注 – コンテナまたは無効なプロジェクトのプロパティだけを変更できます。プロパティを変更するには、その前にプロジェクトが関連付けられているすべてのホストで有効なプロジェクトを無効にする必要があります。変更内容を保存したら、プロジェクトを有効にできます。

▼ プロパティシートを使用してコンテナを変更する

手順 1. **Container Manager** の **GUI** が開いていない場合は、51 ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. コンテナ表示を選択します。

3. コンテナを選択します。

   - コンテナが有効なプロジェクトに使用されていない場合は、ナビゲーションウィンドウで「コンテナ」グループを選択します。右側の区画に「コンテナとグループ」表が表示されます。表でプロジェクトを選択します。

- コンテナが有効なプロジェクトに使用されている場合は、ナビゲーション ウィンドウでコンテナを選択します。必要な場合は、別のコンテナグループをクリックして、個々のコンテナのリストを展開します。「このコンテナに関連付けられたホスト」表が表示され、プロジェクトインスタンスを無効にできます。

注 – プロパティを変更するには、このコンテナを使用するすべてのプロジェクトインスタンスを無効にする必要があります。インスタンスの状態が「有効」の場合は、「このコンテナに関連付けられたホスト」表ですべてのホストを選択し、「無効にする」ボタンを使用してから操作を続けます。

---

4. 右側の区画で「プロパティ」タブを選択します。

   選択したコンテナのプロパティシートが表示されます。次のフィールドを変更できます。

   - 説明 – コンテナの説明を入力します。
   - プロジェクトの種類 – ユーザー、グループ、またはアプリケーションを指定します。
   - UNIX ユーザー名 – 現在の UNIX ユーザー名を変更します。
   - 追加のユーザー – 既存のユーザー名を変更するか、有効な UNIX ユーザー名を追加します。複数の名前を入力する場合は、コンマで区切ります。
   - 追加のグループ – 既存のグループ名を変更するか、有効な UNIX グループ名を追加します。複数の名前を入力する場合は、コンマで区切ります。

注 – 「保存」ボタンを使用できず、テキストフィールドがグレー表示される場合は、コンテナが 1 つまたは複数のコンテナインスタンスで使用されています。「このコンテナに関連付けられたホスト」表に表示されているすべてのホストの状態が「無効」であることを確認します。状態が「有効」のホストがある場合は、無効にする必要があります。

---

5. 「保存」をクリックして変更内容を保存します。

   プロパティシートは表示されたままです。

▼ プロパティシートを使用してプロジェクトを変更する

1 つのプロジェクトのリソースプールまたはリソース予約を変更するには、次の手順に従います。複数のプロジェクトに同じ変更を行うには、97 ページの「リソース変更ジョブによるプロジェクトの変更」を参照してください。

手順 1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. 次のいずれかの方法でプロジェクトインスタンスを選択します。
   - プロジェクトが関連付けられているホストの名前がわかっている場合は、ホスト表示のナビゲーションウィンドウでホスト名を選択します。次に右側の区画で「プロジェクト」タブを選択し、ホストに関連付けられているすべてのプロジェクトを含む表を表示します。
   - プロジェクトの名前がわかっている場合は、コンテナ表示のナビゲーションウィンドウでその名前を選択します。右側の区画に「このコンテナに関連付けられたホスト」表が表示されます。

   注 – プロパティを変更するには、すべてのプロジェクトを無効にする必要があります。状態が「有効」のプロジェクトがある場合は、表の「無効にする」ボタンを使用してから操作を続けます。このボタンは、どちらの表示の表にもあります。

   ---

3. 表でプロジェクトまたはホスト **(**前の手順によって異なります**)** の名前を選択します。

   プロジェクトインスタンスのプロパティシートが表示されます。

4. 変更を行います。

   注 – Solaris 10 では、プロジェクトのリソースプールは変更できません。

   - リソースプール定義 – プロジェクトが関連付けられているプールを変更するには、ドロップダウンメニューから選択します。
   - CPU 予約数 (CPU シェア数) – テキストボックスに新しい値を整数で入力します。
   - メモリーキャップ (M バイト) – テキストボックスに新しい値を入力します。
   - 共有メモリー (M バイト) – テキストボックスに新しい値を入力します。

5. 「保存」をクリックします。

   リソース予約の変更内容が保存されます。

6. **(**省略可能**)** プロジェクトを再度有効にするには、手順 **3** で使用した表に戻り、「有効にする」をクリックします。

リソース変更ジョブによるプロジェクトの変更

リソース変更ジョブ機能を使用して、複数のホストに分散されている複数のプロジェクトのリソース制限を変更できます。これらのプロジェクトは、同じコンテナを使用している必要があります。リソース変更ジョブをすぐに実行して変更を同時に実装するか、あとで実行するようにスケジュールを設定できます。

注 – Solaris 10 では、コンテナが大域ゾーンに配備されている場合にのみ、リソース変更ジョブ機能によってリソース割り当てが変更されます。

注 – CPU 予約数 (CPU シェア数) はすぐに変更されます。メモリーキャップの変更は、スワップに書き込む必要があります。メモリーキャップの変更が大きいと、その調整中にシステムパフォーマンスに影響します。

リソース変更ジョブ表には、次の情報が表示されます。

リソース変更ジョブ名	ジョブ作成時に指定されたジョブ名
ホスト	コンテナが関連付けられているホストの名前
スケジュール	ジョブの実行が予定されている間隔。1 回だけ、毎時、毎日、毎週、毎月のいずれかです。
状態	ジョブの状態。値は、待ち状態、成功、失敗のいずれかです。

次の例では、「Webserver」と「Rollup」という名前のプロジェクトを使用して、リソース変更ジョブ機能を使用して組織全体のシステムリソースを管理する方法を示します。この例では、オンラインストアで Web サイトからの注文を処理します。Webserver プロジェクトは、北米中の Web サーバーで使用されている CPU とメモリーリソースを管理するために作成されました。Rollup プロジェクトは、データベースに必要なリソースを管理するために作成されました。日中と夕方には、ユーザーが Web サイトで注文を行うので、Web サーバーのリソースの需要が高くなります。深夜 0 時を過ぎると、Web サーバーの需要が大幅に下がります。夜の間に、日中の売上に関するレポートを実行するようにデータベースのスケジュールが設定されています。

CPU 8 個、物理メモリーが 6000M バイトのシステムでこの 2 つのコンテナに必要なリソースを管理するには、次の表に示す合計 4 つのリソース変更ジョブを作成できます。

表 **4–2** リソース変更ジョブのスケジュールの例

コンテナ名	リソース変更ジョブ名	開始時刻	間隔	リソース変更
Webserver	webserver-day	午前 6 時	日別	CPU シェア数: 6 メモリー: 2500M バイト

表 4–2 リソース変更ジョブのスケジュールの例　(続き)

コンテナ名	リソース変更ジョブ名	開始時刻	間隔	リソース変更
Rollup	rollup-day	午前 6 時	日別	CPU シェア数: 1 メモリー: 2000M バイト
Webserver	webserver-night	深夜 0 時	日別	CPU シェア数: 1 メモリー: 2000M バイト
Rollup	webserver-night	深夜 0 時	日別	CPU シェア数: 6 メモリー: 2500M バイト

2 つのリソース変更ジョブが毎朝 6 時に実行され、Webserver と Rollup の各プロジェクトのリソースが変更されます。日中は、Web サーバーの需要が高いため、Webserver プロジェクトに CPU と物理メモリーリソースの大半が割り当てられています。毎日、深夜 0 時にほかの 2 つのリソース変更ジョブが実行され、需要の変化に応じてシステムのリソースが再割り当てされます。その日の売上を記録するためにデータベースにリソースが必要になり、Web サーバーの需要は低くなっているので必要なリソースは少なくなります。

この機能は、Sun Management Center のジョブ管理機能に似ていますが、Container Manager の GUI を使用した場合は Container Manager のジョブをすべて管理できます。Sun Management Center のジョブ機能の詳細は、『Sun Management Center 3.6 ユーザーガイド』の「ジョブ管理の概念」を参照してください。

▼ リソース変更ジョブを使用してプロジェクトを変更する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. コンテナ表示を選択します。
3. ナビゲーションウィンドウでプロジェクト名を選択します。
   プロジェクトはホストに関連付けられている必要があります。
4. 右側の区画で「ジョブ」タブを選択します。
   「リソース変更ジョブ」表が表示されます。

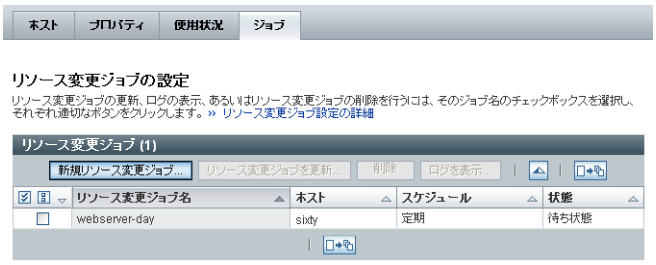


図 **4–3** 例: 「リソース変更ジョブ」表

5. 表の中の「新規リソース変更ジョブ」ボタンをクリックします。
   リソース変更ジョブウィザードが表示されます。「概要」パネルが表示されます。

6. リソース変更ジョブの名前を指定します。説明の入力は省略可能です。
   名前は 32 文字以内で入力します。空白文字、ハイフン (-)、下線 (_)、およびピリオド (.) の各文字は使用できます。空白文字は下線 (_) に変換されます。

   「ホストを選択」パネルが表示されます。選択したコンテナが関連付けられているすべてのホストの名前が「選択可能」リストに表示されます。このウィンドウで 1 つまたは複数のホストを選択し、そのリソース制限を変更できます。

7. 「選択可能」リストでホストを選択し、「追加」をクリックして「選択」リストに追加します。「すべてを選択」を使用してすべてのホストを追加することもできます。
   ホスト名が「選択」フィールドに移動します。

8. 新しい最小 **CPU** 予約数 **(CPU** シェア数**)** を指定します。メモリーキャップは省略可能です。
   新しいリソース制限は、前の手順で選択したすべてのホストに適用されます。

9. リソース変更ジョブの開始日、開始時刻、および間隔を指定します。
   リソース制限の変更内容は、指定した日時に有効になります。

10. 「概要」パネルで設定を確認します。修正するには、「戻る」ボタンをクリックします。変更を完了したら、「完了」をクリックします。
    ウィザードが終了します。ジョブが「ジョブ」表に追加されます。状態は、ジョブの実行が予定されている日時まで「待ち状態」になります。リソース制限の変更内容は、指定した日時に有効になります。

▼ 待ち状態のリソース変更ジョブを変更する

「ジョブ」表で状態が「待ち状態」のジョブを変更するには、次の手順に従います。

手順 1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. ナビゲーションウィンドウでコンテナ表示を選択します。
3. 右側の区画で「ジョブ」タブを選択します。
4. 「リソース変更ジョブ」表で、名前の横のチェックボックスを選択し、変更するジョブを選択します。
   チェックボックスにチェックマークが表示されます。
5. リソース変更ジョブを更新ウィザードを起動するには、「リソース変更ジョブを更新」ボタンをクリックします。
   パネルを移動する詳細な手順については、99 ページの「リソース変更ジョブを使用してプロジェクトを変更する」を参照してください。
6. 変更を完了したら、「完了」をクリックします。
   ウィザードが終了します。ジョブの変更内容は保存されます。

▼ リソース変更ジョブのログを表示する

完了した変更ジョブのログを表示するには、次の手順に従います。ジョブで複数のホストが変更された場合は、ホストごとのジョブの状態がログに記録されます。

手順
1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウでコンテナ表示を選択します。
3. 右側の区画で「ジョブ」タブを選択します。
4. 「リソース変更ジョブ」表で、名前の横のチェックボックスを選択し、ログを表示するジョブを選択します。
   チェックボックスにチェックマークが表示されます。
5. 「ログを表示」ボタンをクリックします。
   リソース変更ジョブのログファイルが表示されます。

プロジェクトの削除

不要になったプロジェクトとそのコンテナは削除できます。プロジェクトとそのコンテナを削除するには、その前に、関連付けられているすべてのホストからプロジェクトを削除する必要があります。プロジェクトとコンテナを削除すると、コンテナがデータベースから削除され、プロジェクト用に収集されたデータが削除されます。プロジェクトのデータはすべてデータベースから削除されるので、削除したプロジェクトの履歴データは確認できなくなります。レコードとすべての履歴データが削除されるので、「削除」はプロジェクトの状態ではありません。

Solaris 8 OS でプロジェクトを削除するには、プロジェクト内で実行中のプロセスが停止している必要があります。

プロジェクトを削除すると、Solaris のバージョンによって、次の処理が実行されます。

Solaris 8 OS	`lnode` が削除されてから、プロジェクトが削除されます。
Solaris 9 OS と Solaris 10 OS	プロジェクト内で実行中のプロセスがデフォルトプロジェクトに移動し、`/etc/project` データベースからエントリが削除されます。

▼ コンテナを削除する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. コンテナに無効または有効なプロジェクトがないことを確認します。
3. ナビゲーションウィンドウでコンテナ表示を選択します。
4. 削除するコンテナを選択します。
5. 「削除」をクリックします。
   コンテナがコンテナ表示から、またデータベースから削除されます。

第 5 章

リソースプールの管理

この章では、リソースプールを削除、使用、および管理する手順を示します。

この章では、以下の内容について説明します。

- 103 ページの「リソースプールの概要」
- 104 ページの「新規リソースプールの作成」
- 107 ページの「リソースプールの削除」

リソースプールの概要

Solaris 9 と Solaris 10 では、リソースプール (プール) は、ホストのリソースを分割するための構成機構です。すべてのホストにはデフォルトプールがあり、プロセスはすべて最初はこのプールにバインドされます。新規に作成するプールでは、デフォルトプールからプロセッサが取得されます。ホストは複数のプールに分割できます。また、`/etc/project` ファイルの各エントリには、関連付けられているプールがあります。`/etc/project` のエントリにプールが指定されていない場合、そのエントリはデフォルトプールにバインドされます。プールが指定されている場合、プロジェクトはそのプールにバインドされます。プロジェクトで新しく開始されたプロセスは、対応するプールに結合されます。

Solaris 8 を使用している場合、使用できるリソースプールは 1 つだけです。コンテナは自動的に `pool_default` に割り当てられます。この OS バージョンでは、リソースプールは存在しないので、見かけ上、`pool_default` が作成されます。Solaris 8 が動作するホスト上の CPU はすべて単一のプールに含まれると見なされます。

ソフトウェアをインストールすると、システム上の既存のリソースプールに関する情報がデータベースに追加されます。Container Manager では、GUI を使用してリソースプールを管理および作成できます。ホスト表示を使用してホスト上のリソースプールを管理します。ナビゲーションウィンドウからプールを表示するには、ホストの横にあるボタンをクリックします。ツリーが展開され、ホスト名の下の対応するプールが表示されます。

注 – リソースプールを管理 (作成、変更、削除) するには、プール管理者権限が必要です。プール管理者は、Solaris Container Manager ソフトウェアを設定する際に指定します。

新規リソースプールの作成

Container Manager ソフトウェアをホストにインストールすると、既存のリソースプールがすべて検出され、モジュールの管理情報ベース (MIB) 表に読み込まれます。新規に作成したプールでは、ホストのデフォルトプール (`pool_default`) からプロセッサが取得されます。デフォルトプールには 1 つの完全な CPU が必ず割り当てられています。したがって、新規プールの作成に使用可能な CPU 数は、使用可能な CPU 数 - 1 です。

「リソースプール」表は、ホスト表示の「リソースプール」タブにあります。1 つのプールに関する情報が 1 行に表示されます。各リソースプールについて、次の情報が表示されます。

リソースプール名	リソースプールの名前
CPU	このプールに含まれる CPU の総数
予約されていない CPU 数	コンテナの作成に使用可能な、または現在、使用可能な CPU 数
CPU シェア数	ゾーンまたはプロジェクトに分散できるリソースプール内の CPU シェア数

▼ 新しいリソースプールを作成する

手順 1. **Container Manager** の **GUI** が開いていない場合は、51 ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. ナビゲーションウィンドウで左側のタブをクリックしてホスト表示を選択します。
   ホストのリストがナビゲーションウィンドウに表示されます。

3. ホスト名を選択します。
   右側の区画に「リソースプール」表が表示されます。

4. 「新規リソースプール」ボタンをクリックします。
   「新規リソースプール」ダイアログが表示されます。ホスト上の CPU の総数と使用可能な CPU 数が表示されます。

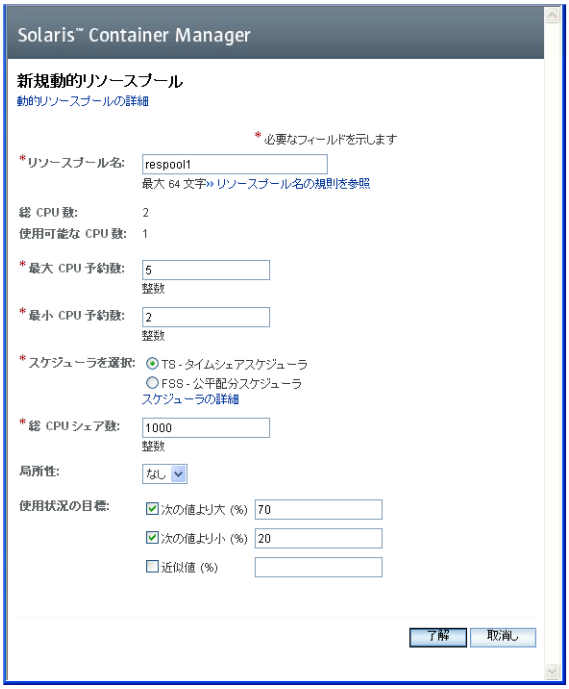


図 **5–1** Solaris 10 のホストの「リソースプール」ウィンドウ

注 **–** ダイアログは、オペレーティングシステムによって異なります。Solaris 10 のホストを使用している場合は、オプションが多くなります。

---

5. 「リソースプール名」フィールドに新規プールの名前を指定します。
6. **(Solaris 8** と **9** のみ**)** プールの **CPU** 数を指定します。
   この数値は、「使用可能な CPU 数」フィールドの数値以下の整数である必要があります。
7. タイムシェア **(TS)** スケジューラまたは公平配分スケジューラ **(FSS)** を選択します。
   スケジューラについては、69 ページの「公平配分スケジューラ (FSS)」および 69 ページの「タイムシェアスケジューラ (TS)」 を参照してください。

8. **CPU** シェアの総数を入力します。

   CPU シェアの総数は、プールのリソースの割合をプロジェクトに割り当てる基本単位で、整数である必要があります。たとえば、プールに 100 CPU シェアを割り当てた場合、プールの 50 シェアを 1 つのプロジェクトに、20 シェアを別のプロジェクトに、30 をさらに別のプロジェクトに割り当てることができます。

9. **(Solaris 10** のみ**)** 最大と最小の **CPU** 予約数を入力します。

   最小 CPU 予約数は、プールに必要な最小 CPU 数です。最大 CPU 予約数は、プールに割り当てることができる最大 CPU 数です。

   最小 CPU 数は、(CPU の総数) - (ほかのプールの最小 CPU 数) です。最大 CPU 数に制限はありません。

   動的リソースプールの割り当ての詳細は、『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』を参照してください。

10. **(Solaris 10** のみ **-** 省略可能**)** プールの動的割り当ての局所性を設定します。

    「なし」は、ボード上の CPU の物理位置を考慮しないことを指定します。

    「疎」は、ボード上の CPU の物理位置を考慮することを指定します。

    「密」は、CPU がローカル、つまりプールに割り当てられた最初の CPU と同じボード上にある必要があることを指定します。

11. **(Solaris 10** のみ **-** 省略可能**)** 使用状況の目標を設定します。

    次のオプションがあります。

    - 「次の値より大」フィールドまたは「次の値より小」フィールドに 0 ～ 100 の間の整数を入力するか、両方のフィールドに入力して範囲を指定します。
    - 「近似値」フィールドに 0 ～ 100 の間の整数を入力します。

12. 「了解」をクリックして、変更を保存します

    「リソースプール」表に、新規プールの情報が追加されます。

リソースプールの変更

CPU、スケジューラの種類、または CPU シェア数を変更するときは、リソースプールを変更します。Solaris 10 を使用している場合は、プールの最大と最小の CPU 予約数、局所性、および使用状況の目標も変更できます。

注 – デフォルトのリソースプールは変更できません。

▼ リソースプールを変更する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで左側のタブをクリックしてホスト表示を選択します。
   ホストのリストがナビゲーションウィンドウに表示されます。
3. リストでホストを探し、名前の横の三角形の展開ボタンをクリックします。
   ツリーが展開し、そのホストにバインドされているリソースプールがすべて表示されます。
4. ナビゲーションウィンドウで名前をクリックしてリソースプールを選択します。
5. 「プロパティ」タブを選択します。
6. 必要な属性を編集します。
7. 「保存」をクリックします。

リソースプールの削除

ホストのデフォルトのリソースプールは削除できません。コンテナが関連付けられているリソースプールは削除できません。

▼ リソースプールを削除する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで左側のタブをクリックしてホスト表示を選択します。
   ホストのリストがナビゲーションウィンドウに表示されます。
3. ホスト名を選択します。
   右側の区画に「リソースプール」表が表示されます。

4. 「削除」ボタンを使用可能にするには、リソースプールの横のチェックボックスを選択します。

   「削除」ボタンが使用可能にならない場合は、このプールを使用しているコンテナがあるため削除できません。

5. 「削除」ボタンをクリックします。

   選択したリソースプールが削除され、名前がツリーから削除されます。

第 6 章

ゾーンの管理

この章では、Solaris 10 システムでゾーンを作成、使用、および管理する手順を示します。

この章では、以下の内容について説明します。

- 109 ページの「ゾーンの概要」
- 110 ページの「非大域ゾーンの状態」
- 110 ページの「非大域ゾーンの作成」
- 116 ページの「非大域ゾーンのコピー」
- 117 ページの「非大域ゾーンの削除、起動、停止」

ゾーンの概要

Solaris 10 の機能であるゾーンは、アプリケーションを実行する安全で独立した環境です。ゾーンを使用すると、Solaris のインスタンス内で仮想化されたオペレーティングシステム環境を作成できます。ゾーンを使用すると、1 つまたは複数のプロセスを、システムのほかのプロセスから独立して実行できます。たとえば、ゾーン内で実行されているプロセスは、ユーザー ID やその他の資格情報に関係なく、同じゾーン内のほかのプロセスだけにシグナルを送信できます。エラーが発生した場合は、ゾーン内で実行されているプロセスだけに影響します。

ゾーンには、独自の IP アドレス、ファイルシステム、一意の root ユーザー名とパスワードファイル、およびネームサーバーを設定できます。

すべての Solaris 10 システムには大域ゾーンが含まれます。大域ゾーンは、システムのデフォルトのゾーンで、システム全体の管理に使用されます。大域ゾーンは、設定、インストール、またはアンインストールできません。

システムに作成できるゾーン数の上限は 8192 です。1 つのシステムで効率的にホストできるゾーン数は、すべてのゾーンで実行されるアプリケーションソフトウェアに必要な総リソース量によって決まります。

Container Manager では、非大域ゾーンを作成、削除、変更、コピー、停止、および再起動できます。Container Manager では、既存のゾーンを検出したり、ゾーンの変更を検出したり、ゾーンの CPU、メモリー、およびネットワークの使用状況を監視および保管したり、ゾーンの上下のアラームを生成したりできます。

注 – 非大域ゾーンの管理 (作成、変更、コピー、削除、起動、停止) は、ゾーン管理者だけが行うことができます。ゾーン管理者は、Solaris Container Manager ソフトウェアを設定する際に指定します。

ゾーンの詳細については、『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』の第 16 章 「 Solaris ゾーンの紹介」 を参照してください。

非大域ゾーンの状態

Container Manager を使用して非大域ゾーンを作成できます。

非大域ゾーンには、次の状態があります。

- 構成済み
- 不完全
- インストール済み
- 準備完了
- 稼働
- 停止
- ダウン

ゾーンの状態の詳細は、『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』を参照してください。

注 – 大域ゾーンは、つねに「稼働」の状態です。

非大域ゾーンの作成

非大域ゾーンを作成し、このゾーン内で実行中のアプリケーションをほかのアプリケーションから分離できます。

始める前に

使用可能な CPU シェアがあるリソースプールが必要です。新しいリソースプールを作成する方法については、104 ページの「新規リソースプールの作成」を参照してください。

▼ 非大域ゾーンを作成する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで「ホスト」タブを選択してホスト表示を選択します。
   ホストのリストがナビゲーションウィンドウに表示されます。
3. **Solaris 10** のホストを選択します。
4. 「ゾーン」タブを選択します。
5. 「新規ゾーン」ボタンをクリックします。
   「新規ゾーン」ウィザードが表示されます。

図 **6–1** ゾーン作成時のパラメータのウィンドウ

6. 各フィールドに、ゾーン名、ゾーンのホスト名、ゾーンパス、**IP** アドレス、およびネットワークインタフェースの適切な値を入力します。

   ゾーンのホスト名は、仮想ホストとしてのゾーンの一意の名前です。ホスト名やマシン名ではありません。

   ゾーンパスは、ルート (/) ディレクトリを基準とした絶対パスです。

   注 – 「ゾーンのパス」フィールドに指定したディレクトリが存在する場合は、ルートディレクトリのアクセス権が 700 でなければ、ゾーンは作成されません。

   「ゾーンのパス」フィールドに指定したディレクトリが存在しない場合は、Solaris Container Manager によってこの名前のディレクトリが作成され、ルートディレクトリに 700 のアクセス権が割り当てられます。

   各ゾーンには、ゾーンが「インストール済み」の状態から「準備完了」の状態に移行したときにネットワークインタフェースを設定できます。

7. **(**省略可能**)** システムの再起動時にゾーンを自動的に再起動するには、「有効」チェックボックスを選択します。

8. **root** ユーザー名を入力します。

図 **6–2** ゾーンの属性のウィンドウ

9. **(**省略可能**)** 「ロケール」、「端末の種類」、「ネームサービス」、および「タイムゾーン」の各リストで適切な値を選択します。

   ロケールは、このゾーンで使用する言語です。

   端末の種類は、端末の種類です。

   ネームサービスは、ホスト名と IP アドレス間の変換を行います。

   タイムゾーンは、ゾーンのタイムゾーンです。

10. ゾーン用のリソースプールを選択します。

11. 「**CPU** シェア数」フィールドと「プロジェクトの **CPU** シェア数」フィールドに適切な値を入力します。

図 **6–3** CPU シェア数のウィンドウ

「CPU シェア数」フィールドには、リソースプールからこのゾーンに割り当てる CPU シェア数を指定します。この数は、リソースプールで使用可能な CPU シェア数以下である必要があります。

「プロジェクトの CPU シェア数」フィールドには、ゾーン内のプロジェクトに割り当てる CPU シェア数を指定します。

12. **(省略可能) IPQoS (IP quality of service)** 機能の最小の入力帯域幅と出力帯域幅を入力します。

図 **6–4** IPQoS の属性のウィンドウ

13. ゾーンの追加属性を設定するかどうかを選択します。

- ゾーンの追加属性を設定する場合は、「はい」を選択して「次へ」をクリックします。

  このゾーンの追加 IP アドレスを入力し、適切なネットワークインタフェースを選択します。

  このゾーンに設定するデバイスディレクトリを入力します。

  このゾーンにマウントする追加ファイルシステムの詳細を入力します。

  このゾーンが大域ゾーンと共有するパッケージソフトウェアがあるディレクトリを入力します。

  これらの追加の属性の詳細は、『Solaris のシステム管理 (Solaris コンテナ : 資源管理と Solaris ゾーン)』の「ゾーン構成データ」を参照してください。

  設定を確認し、「完了」をクリックして変更内容を保存します。

- ゾーンの追加属性を設定しない場合は、「いいえ」を選択して「次へ」をクリックします。

  設定を確認し、「完了」をクリックして変更内容を保存します。

ゾーン表に、新規ゾーンの情報が追加されます。ゾーンを作成すると、「未完了」の状態になります。大域ゾーンから特定のパッケージを継承したら、ゾーンは「稼働」の状態に変わります。

非大域ゾーンのコピー

非大域ゾーンをコピーすると、そのプロパティが新しいゾーンに継承されます。

▼ 非大域ゾーンをコピーする

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ナビゲーションウィンドウで「ホスト」タブを選択してホスト表示を選択します。ホストのリストがナビゲーションウィンドウに表示されます。
3. **Solaris 10** のホストを選択します。
4. 「ゾーン」タブを選択します。
5. 非大域ゾーン名の横のオプションボタンを選択します。
6. 「ゾーンのコピー」ボタンをクリックします。
7. 適切なパラメータを入力し、「了解」をクリックします。

図 **6–5** ゾーンのコピーのウィンドウ

ゾーンのコピー時には、次の属性がデフォルト値に設定されます。

属性	デフォルト値
端末タイプ	dtterm
ロケール	C
タイムゾーン	GMT

これらの属性のデフォルト値は、ゾーンをコピーし、「稼働」の状態になったときに変更できます。

ゾーン表に、新規ゾーンの情報が追加されます。ゾーンをコピーすると、「構成済み」の状態になります。大域ゾーンから特定のパッケージを継承したら、ゾーンは「稼働」の状態に変わります。

非大域ゾーンの削除、起動、停止

ゾーンの状態は、必要に応じて変更できます。

▼ 非大域ゾーンを削除、起動、または停止する

手順 1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. ナビゲーションウィンドウで「ホスト」タブを選択してホスト表示を選択します。
   ホストのリストがナビゲーションウィンドウに表示されます。
3. **Solaris 10** のホストを選択します。
4. 「ゾーン」タブを選択します。
5. 非大域ゾーン名の横のオプションボタンを選択します。
6. 「削除」、「起動」、または「停止」のいずれかのボタンをクリックします。
   ゾーンを起動すると、「インストール済み」の状態から「稼働」の状態に変わります。ゾーンを停止すると、「稼働」の状態から「インストール済み」の状態に変わります。

ゾーンのログファイルの表示

▼ ゾーンのログファイルを表示する

手順 1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。

2. ナビゲーションウィンドウで「ホスト」タブを選択してホスト表示を選択します。
   ホストのリストがナビゲーションウィンドウに表示されます。
3. **Solaris 10** のホストを選択します。
4. 「ゾーン」タブを選択します。
5. 「ゾーン」表の下にあるフィールドにゾーンの名前を入力します。
6. 「ゾーンのログ」ボタンをクリックします。
   このゾーンのログファイルが表示されます。

第 7 章

アラームの管理

この章では、アラームを管理する方法について説明します。

アラームの管理

注 – アラームを管理するには Sun Management Center 3.6 の Java コンソールレイヤをインストールする必要があります。詳細は、『Sun Management Center 3.6 ユーザーガイド』の第 12 章「アラームの管理」を参照してください。ユーザーがほかの製品を使用してアラームを管理している可能性もあります。

アラームの設定

CPU またはメモリーが、事前に設定したしきい値に達したときに警告するようにアラームを設定できます。これらのアラームは、リソース予約をもっと適切なレベルに調整するために必要なフィードバック情報を提供します。

コンテナによる CPU と物理メモリーのリソースの使用に対してアラームのしきい値を設定できます。アラームには、重大、メジャー、およびマイナーの 3 レベルがあります。アラームの生成時に電子メールが送信されるように設定することもできます。アラームは、ナビゲーションウィンドウと「コンテナ」テーブルでアイコンとして表示されます。このアイコンの上にカーソルを置くと、アラームの詳細を示すツールチップが表示されます。

Container Manager の GUI では、そのモジュールで生成されたアラームだけが表示されます。Container Manager モジュールで生成されたアラームは、Sun Management Center の Java コンソールと Web コンソールに表示されます。Sun Management Center を使用して Container Manager のアラームを表示した場合、アラームのしきい値の名前は次のように対応します。

表 7–1 アラームのしきい値の名前

Container Manager	Sun Management Center
重大	重大
メジャー	警告
マイナー	情報

▼ アラームのしきい値を設定する

手順

1. **Container Manager** の **GUI** が開いていない場合は、51 ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ホスト表示で、コンテナが関連付けられているホストを選択します。
3. 「プロジェクト」タブを選択します。
   このホストに関連付けられているすべてのコンテナを示す表が表示されます。
4. 表の「プロジェクト名」列で、アラームを設定するプロジェクトを選択します。
   コンテナの「プロパティ」ページが表示されます。
5. 「アラームしきい値」タブを選択します。
   「アラームしきい値」ページが表示されます。アラームには、重大、メジャー、およびマイナーの 3 レベルがあります。
6. 設定するアラームのレベルを探し、テキストフィールドにアラームの値を入力します。
   次の 3 つのアラーム設定があります。
   - CPU アラームしきい値 (次の値より小) – 整数または小数を指定します。CPU の使用量がこの値を下回ったときにアラームが発生します。
   - メモリーアラームしきい値 (次の値より大) – M バイト単位の整数を指定します。メモリーの使用量がこの値を上回ったときにアラームが発生します。
   - 送信先 – 有効な電子メールアドレスを指定します。アラームが発生したときに、電子メール警告がこのアドレスに送信されます。

   注 – 1 つ、2 つ、または 3 つのレベルのアラームを一度に設定できます。複数のアラームが発生した場合は、もっとも値が大きいアラームがアイコンとして GUI に表示されます。同様に、もっとも値が大きいアラームの電子メール警告が送信されます。
7. 「保存」をクリックします。
   これでアラームは設定されます。Container Manager によって、このコンテナがアラームの値に対して監視されます。

アラームの生成時に実行する処理を追加できます。Sun Management Center 3.6 の Java コンソールレイヤを使用して処理を追加できます。たとえば、ゾーンが実行されているホストが応答しなかったときに、Container Manager に重大アラームを生成させることができます。このアラームは、重大アラームの処理が設定されているホストに伝播されます。

▼ アラームのしきい値を削除する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. ホスト表示で、コンテナが関連付けられているホストを選択します。
3. 「プロジェクト」タブを選択します。
   このホストに関連付けられているすべてのコンテナを示す表が表示されます。
4. 表の「プロジェクト名」列で、アラームを削除するプロジェクトを選択します。
   コンテナの「プロパティ」ページが表示されます。
5. 「アラームしきい値」タブを選択します。
   「アラームしきい値」ページが表示されます。コンテナに設定されているアラームの値が表示されます。
6. テキストフィールドからアラームの値を削除します。
7. 「保存」をクリックします。
   これでアラームの設定が解除されます。

▼ 未確認のアラームを表示する

手順

1. **Container Manager** の **GUI** が開いていない場合は、**51** ページの「**Container Manager** の **GUI** を起動する」に従って起動します。
2. 「未確認のアラーム」タブを選択します。
   「未確認のアラーム」表に、Container Manager モジュールで現在、未確認のアラームがすべて表示されます。

   「未確認のアラーム」表には、アラームの重要度、アラームのメッセージ、エージェント名またはホスト名 (管理対象オブジェクト)、アラームの開始時刻、およびアラームが Sun Management Center コンソールで確認されたかどうかが表示されます。
3. 「管理対象オブジェクト」列で適切なリンクをクリックし、アラーム状態のオブジェクトを表示します。

第 8 章

レポートの作成と拡張アカウンティングデータの使用

この章では、レポートを作成する方法と、拡張アカウンティングデータを使用する方法について説明します。

この章では、以下の内容について説明します。

- 123 ページの「レポート」
- 126 ページの「データ収集プロセス」
- 126 ページの「レポート要求」

レポート

Performance Reporting Manager ソフトウェアがインストールされている場合は、コンテナ、ホスト、またはリソースプールごとの CPU とメモリーのリソース使用量の詳細を示すレポートを生成できます。ゾーンについては、CPU、メモリー、入力帯域幅、および出力帯域幅の詳細を示すレポートを生成できます。これらのグラフレポートは、すべて Container Manager の GUI の「使用状況」タブから生成できます。グラフレポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用できます。この待ち時間は、グラフレポートに使用するデータを収集し、データベースに保存するために必要です。週別グラフと月別グラフの待ち時間は、24 ～ 48 時間です。

次の 2 種類の使用状況レポートを生成できます。

- 総稼働率レポート。このレポートは、すべてのリソースオブジェクトのリソース使用状況を示します。
- 累積稼働率レポート。このレポートは、上位 5 つのリソースオブジェクトのリソース使用状況を示します。

表 **8–1** 使用状況レポートの種類

リソース オブジェクト	総稼働率レポート	累積稼働率レポート
ホスト	すべてのプロジェクトの使用状況	上位 5 つのリソースオブジェクトの使用状況。 Solaris 10 では、リソースオブジェクトはリソースプール、ゾーン、またはプロジェクトです。 Solaris 8、9 では、リソースオブジェクトはリソースプールまたはプロジェクトです。
プール	すべてのプロジェクトの使用状況	上位 5 つのリソースオブジェクトの使用状況。 Solaris 10 では、リソースオブジェクトはゾーンまたはプロジェクトです。 Solaris 8、9 では、リソースオブジェクトはプロジェクトです。
ゾーン	すべてのプロジェクトの使用状況	上位 5 つのプロジェクトの使用状況。 Solaris 10 では、リソースオブジェクトはプロジェクトです。
コンテナ	すべてのプロジェクトの使用状況	上位 5 つのホストの使用状況。
プロジェクト	プロジェクトの使用状況	該当なし

Container Manager の GUI から、次の種類のリソース使用状況のグラフレポートを生成できます。

コンテナ　返されるデータは、有効なコンテナと無効なコンテナを含むすべてのコンテナのリソース使用量の平均です。無効なコンテナについては履歴データが表示されます。履歴データから、コンテナでリソース消費が効果的に制御されているかどうかがわかります。データは、すべての有効なコンテナの最小の CPU とメモリーキャップのリソース予約に対する割合で表されます。この割合は、リソース予約量と、実際のリソース使用量の比較です。

有効なコンテナ　返されるデータは、選択した有効なコンテナで現在使用されている CPU 数とメモリー容量です。

コンテナグループ　返されるデータは、選択したグループ内のすべてのコンテナのリソース使用量の平均です。この割合は、選択したコンテナのリソース予約量と、実際のリソース使用量の比較です。

ホスト	返されるデータは、選択したホスト上のすべての有効なコンテナの総計です。
ホストグループ	返されるデータは、グループ内のすべてのホストのリソース使用量の平均です。データは、ホストのリソース総量に対する割合で表されます。
リソースプール	返されるデータは、選択したリソースプール内のすべての有効なコンテナの総計です。
ゾーン	返されるデータは、選択したゾーン内のすべての有効なプロジェクトの総計です。

複数の異なるホスト上にある複数のコンテナのグラフの場合、返されるデータは各ホストの使用割合の平均です。

レポートのデータは、有効なコンテナ、ゾーン、リソースプール、またはホストのコンマ区切り (CSV) 形式のテキストファイルにエクスポートすることもできます。このテキストファイルは、たとえば請求や会計のアプリケーションのインタフェースファイルとして使用できます。CSV 形式のレポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用できます。この待ち時間に、CSV レポートに使用するデータが収集され、データベースに保存されます。エクスポートされるデータは、グラフレポートに表示されるデータよりも具体的です。CSV レポートには、過去 24 時間のデータが含まれます。

エクスポートした CSV レポートには、次の情報が含まれます。

ホスト名	コンテナが関連付けられているホストの名前
タイムスタンプ	レコードの日時
CPU 予約数	コンテナの CPU 予約数
CPU 使用	コンテナ内のすべてのプロセスの総 CPU 使用数
使用 CPU vs 予約 CPU	CPU 予約数に対する CPU 使用数 (%)
CPU 拡張アカウンティング情報	CPU の拡張アカウンティング情報
メモリーキャップ	物理メモリーキャップ
メモリー使用	物理メモリーの使用量
使用されているメモリーの割合	ホストで使用されている物理メモリーの割合 (%)
使用メモリー vs 予約メモリー	メモリーの予約量に対するメモリー使用量 (%)
コンテナプロジェクト ID	コンテナのプロジェクト ID
プロジェクト名	コンテナのプロジェクト名

データ収集プロセス

Container Manager では、サーバー層にある Performance Reporting Manager のデータ収集サービスが使用されます。このデータ収集サービスでは、エージェント層にある Sun Management Center の履歴ログ機能が使用されます。サーバー層のデータ収集サービスでは、エージェントマシンからデータが収集され、データベースに保存されます。Performance Reporting Manager で収集されたデータは、事前に定義された間隔で集約 (「ロールアップ」) されます。データの最小値、最大値、および平均値が計算され、時間別データ、週別データ、および月別データとして保存されます。最後に、同じ期間の最小値、最大値、および平均値が計算されます。

Container Manager で生成されたレポートには、レポートの要求パラメータによって、このうち任意のデータを含めることができます。Performance Reporting Manager のデータ収集方法の詳細は、『Sun Management Center 3.6 Performance Reporting Manager User's Guide』の「Data Collection Process」を参照してください。

レポート要求

ホスト、ホストグループ、コンテナ、コンテナグループ、リソースプール、ゾーン、および有効なコンテナについて、CPU とメモリーの両方のリソース使用状況レポートを使用できます。レポートを表示する前に、ブラウザのキャッシュを毎回更新するように設定します。6 種類のレポートの一覧は、123 ページの「レポート」を参照してください。次の期間に使用された CPU リソースとメモリーリソースのレポートを生成できます。

- 日別 – 過去 24 時間のデータを 1 時間 1 ポイントで示します。
- 週別 – 過去 7 日間からのデータを 1 日 1 ポイントで示します。
- 月別 – 過去 30 日間からのデータを 1 日 1 ポイントで示します。
- リアルタイム - 1 ポイントで 30 秒ごとのデータを示します。有効なプロジェクトについてのみ生成できます。

日別グラフレポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用可能になります。データが収集され、データベースに保存されてから、レポートが作成されます。履歴データに基づいた、無効なコンテナやコンテナの CPU とメモリーのリソース使用状況レポートも表示できます。

使用されている CPU リソースとメモリーリソースのリアルタイムのレポートは、有効なプロジェクトについてのみ生成できます。

▼ ホストのリソース使用状況レポートを要求する

ホストの日別、週別、または月別のレポートを生成するには、次の手順に従います。

手順 **1.** ブラウザのキャッシュを毎回更新するように設定します。

2. ホスト表示のナビゲーションウィンドウからホストを選択します。

3. 「使用状況」タブを選択します。
過去 24 時間のホストの CPU とメモリーの使用量が表示されます。

注 – Solaris 10 では、CPU およびメモリーの使用量が表示されます。

---

4. 「グラフのプロパティ」のリンクをクリックします。

5. ホスト全体のレポートを生成するか、ホスト内のリソースオブジェクトだけのレポートを生成するかを選択します。

注 – ホストのリソースオブジェクトは、リソースプールまたはプロジェクトです。

Solaris 10 では、ホストのリソースオブジェクトは、リソースプール、ゾーン、またはプロジェクトです。

- ホスト全体のレポートを生成するには、「総稼働率」オプションを選択します。
  「期間」リストから適切な期間を選択します。
  「グラフを更新」ボタンをクリックします。
  CPU とメモリーのリソース使用状況グラフが表示されます。
- ホスト内のリソースオブジェクトのレポートを生成するには、「オブジェクトごとの累積稼働率」オプションを選択します。
  適切なリソースオブジェクトを選択します。
  「期間」リストから適切な期間を選択します。
  「グラフを更新」ボタンをクリックします。
  使用量が多い上位 5 つのリソースオブジェクトの CPU とメモリーのリソース使用状況グラフが表示されます。

6. **(省略可能) 過去 24 時間のデータを CSV ファイルにエクスポートするには、**「データをエクスポート」ボタンをクリックします。
CSV レポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用可能になります。データが収集され、データベースに保存されてから、レポートが作成されます。このデータはグラフでプレビューできません。

エクスポートされるデータは、過去 24 時間のコンテナの時間別データです。したがって、日別グラフのデータとは異なります。

▼ 有効なプロジェクトのリソース使用状況レポートを要求する

有効なプロジェクトの日別、週別、月別、またはリアルタイムのレポートを生成するには、次の手順に従います。

手順 **1.** ブラウザのキャッシュを毎回更新するように設定します。

2. 「ホスト」ビューのナビゲーションウィンドウからホストを選択します。

3. 右側の区画で「プロジェクト」タブを選択します。
このホスト上のプロジェクトのリストが表示されます。

4. レポートを生成する有効なプロジェクトを選択します。

5. 「使用状況」タブを選択します。
過去 24 時間の有効なプロジェクトの CPU とメモリーの使用量が表示されます。

6. 「グラフのプロパティ」のリンクをクリックします。

7. 「総稼働率」オプションを選択します。

8. 「期間」リストから適切な期間を選択します。
有効なプロジェクトのリアルタイムのレポートを生成することもできます。

9. 「グラフを更新」ボタンをクリックします。
CPU とメモリーのリソース使用状況グラフが表示されます。

10. (省略可能**)** 過去 **24** 時間のデータを **CSV** ファイルにエクスポートするには、「データをエクスポート」ボタンをクリックします。

CSV レポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用可能になります。データが収集され、データベースに保存されてから、レポートが作成されます。このデータはグラフでプレビューできません。

エクスポートされるデータは、過去 24 時間のコンテナの時間別データです。したがって、日別グラフのデータとは異なります。

▼ コンテナのリソース使用状況レポートを要求する

コンテナの CPU とメモリーの使用状況レポートを要求するには、次の手順に従います。データは履歴データに基づき、コンテナに基づく有効なコンテナで使用されたリソースの平均です。

手順 **1.** ブラウザのキャッシュを毎回更新するように設定します。

2. コンテナ表示でコンテナを選択します。

3. 「使用状況」タブを選択します。

過去 24 時間のコンテナの CPU とメモリーの使用量が表示されます。

4. 「グラフのプロパティ」のリンクをクリックします。

5. コンテナ全体のレポートを生成するか、コンテナ内のリソースオブジェクトのレポートを生成するかを選択します。

注 – コンテナのリソースオブジェクトはホストです。

- コンテナ全体のレポートを生成するには、「総稼働率」オプションを選択します。

  「期間」リストから適切な期間を選択します。

  「グラフを更新」ボタンをクリックします。

  CPU とメモリーのリソース使用状況グラフが表示されます。
- コンテナ内のリソースオブジェクトのレポートを生成するには、「オブジェクトごとの累積稼働率」オプションを選択します。

  適切なリソースオブジェクトを選択します。

  「期間」リストから適切な期間を選択します。

  「グラフを更新」ボタンをクリックします。

  使用量が多い上位 5 つのリソースオブジェクトの CPU とメモリーのリソース使用状況グラフが表示されます。

▼ リソースプールのリソース使用状況レポートを要求する

リソースプールの日別、週別、または月別のレポートを生成するには、次の手順に従います。

手順 **1.** ブラウザのキャッシュを毎回更新するように設定します。

2. ホスト表示で、リソースプールがバインドされているホストを選択します。
   このホストにバインドされているすべてのリソースプールのリストが、右側の区画の「リソースプール」表に表示されます。
3. 表内でリソースプールの名前を選択します。
   このリソースプールにバインドされているすべてのコンテナを含む表が表示されます。
4. 「使用状況」タブを選択します。
   過去 24 時間のリソースプールの CPU とメモリーの使用量が表示されます。

注 – Solaris 10 では、CPU およびメモリーの使用量が表示されます。

---

5. 「グラフのプロパティ」のリンクをクリックします。
6. リソースプール全体のレポートを生成するか、リソースプール内のリソースオブジェクトだけのレポートを生成するかを選択します。

注 – リソースプールのリソースオブジェクトはプロジェクトです。

Solaris 10 では、リソースプールのリソースオブジェクトはゾーンおよびプロジェクトです。

- リソースプール全体のレポートを生成するには、「総稼働率」オプションを選択します。
  「期間」リストから適切な期間を選択します。
  「グラフを更新」ボタンをクリックします。
  CPU とメモリーのリソース使用状況グラフが表示されます。また、使用状況レポートおよび帯域幅レポートが生成されます。
- リソースプール内のリソースオブジェクトのレポートを生成するには、「オブジェクトごとの累積稼働率」オプションを選択します。
  適切なリソースオブジェクトを選択します。
  「期間」リストから適切な期間を選択します。
  「グラフを更新」ボタンをクリックします。
  使用量が多い上位 5 つのリソースオブジェクトの CPU とメモリーのリソース使用状況グラフが表示されます。また、使用状況レポートおよび帯域幅レポートが生成されます。

7. **(省略可能)** 過去 **24** 時間のデータを **CSV** ファイルにエクスポートするには、「データをエクスポート」ボタンをクリックします。

   CSV レポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用可能になります。データが収集され、データベースに保存されてから、レポートが作成されます。このデータはグラフでプレビューできません。

   エクスポートされるデータは、過去 24 時間のコンテナの時間別データです。したがって、日別グラフのデータとは異なります。

▼ (Solaris 10 のみ) ゾーンのリソース使用状況レポートを要求する

ゾーンの日別、週別、または月別のレポートを生成するには、次の手順に従います。

手順 1. ブラウザのキャッシュを毎回更新するように設定します。

2. ホスト表示のナビゲーションウィンドウから **Solaris 10** のホストを選択します。

3. 「ゾーン」タブを選択します。
   このホスト上のゾーンのリストが表示されます。

4. レポートを生成するゾーンを選択します。

5. 「使用状況」タブを選択します。
   過去 24 時間のゾーンの CPU 、メモリー、および帯域幅の使用量が表示されます。

6. ゾーン全体のレポートを生成するか、ゾーン内のリソースオブジェクトだけのレポートを生成するかを選択します。

   注 – ゾーンのリソースオブジェクトはプロジェクトです。

   - ゾーン全体のレポートを生成するには、「総稼働率」オプションを選択します。

     「期間」リストから適切な期間を選択します。

     「グラフを更新」ボタンをクリックします。

     CPU とメモリーのリソース使用状況グラフが表示されます。
   - ゾーン内のリソースオブジェクトのレポートを生成するには、「オブジェクトごとの累積稼働率」オプションを選択します。

     リストで「プロジェクト」を選択します。

     「期間」リストから適切な期間を選択します。

「グラフを更新」ボタンをクリックします。

使用量が多い上位 5 つのリソースオブジェクトの CPU とメモリーのリソース使用状況グラフが表示されます。

7. **(**省略可能**)** 過去 **24** 時間のデータを **CSV** ファイルにエクスポートするには、「データをエクスポート」ボタンをクリックします。

   CSV レポートは、Performance Reporting Manager ソフトウェアのインストールの 2 時間後から使用可能になります。データが収集され、データベースに保存されてから、レポートが作成されます。このデータはグラフでプレビューできません。

   エクスポートされるデータは、過去 24 時間のコンテナの時間別データです。したがって、日別グラフのデータとは異なります。

付録 A

コマンド行インストール

この付録では、コマンド行から Solaris Container Manager 3.6 (Container Manager) ソフトウェアをインストール、設定、および削除する手順を示します。インストールウィザードの代わりにコマンド行を使用してインストールを行えます。

ソフトウェアのインストール、設定、使用に関する最新の情報は、『Solaris Container Manager 3.6 リリースノート』を参照してください。

インストールウィザードを使用する方法については、第 2 章を参照してください。

- 133 ページの「Container Manager ソフトウェアのインストール」
- 136 ページの「Container Manager の設定」
- 139 ページの「Container Manager ソフトウェアの削除」

Container Manager ソフトウェアのインストール

コマンド行からの Container Manager のインストールは、Sun Management Center アドオンソフトウェアの標準のインストール手順に従います。コマンド行でのアドオンソフトウェアのインストールの詳細は、『Sun Management Center 3.6 インストールと構成ガイド』の付録 B「コマンド行によるアンインストール、インストール、および設定」を参照してください。

Container Manager は、Sun Management Center 3.6 の次のレイヤにインストールする必要があります。

- サーバーレイヤ
- エージェントレイヤ (レポートを生成する全ホスト上)

対話形式の es-inst インストールスクリプトでは、適切な Container Manager ソフトウェアパッケージがサーバーレイヤとエージェントレイヤにインストールされます。また、すべてのレイヤに対して、パッケージ、モジュール設定ファイル、およびライ

ブラリが、Sun Management Center 3.6 の標準の場所にインストールされます。サポート対象のプラットフォームおよびインストール要件の詳細は、31 ページの「Container Manager ソフトウェアについて」を参照してください。

Container Manager ソフトウェアは、次のときにインストールできます。

- Sun Management Center 3.6 ソフトウェアと同時にインストールします。
- 既存の Sun Management Center のアップグレード手順の一環としてインストールできます。
- Sun Management Center 3.6 のインストール、または Sun Management Center 3.6 へのアップグレードの完了後。この場合、Container Manager ソフトウェアを別個にインストールします。

Container Manager のパフォーマンスとアカウンティングデータ機能を使用するには、Performance Reporting Manager ソフトウェアもインストールする必要があります。コマンド行でのこのソフトウェアのインストールおよび設定の詳細は、『Sun Management Center 3.6 Performance Reporting Manager User's Guide』の付録 B「Installing Using the Command Line」を参照してください。これらの機能を使用しない場合は、このアドオンソフトウェアをインストールする必要はありません。

▼ Sun Management Center のインストール時に Container Manager をインストールする

手順

- コマンド行での具体的な手順については『**Sun Management Center 3.6** インストールと構成ガイド』の付録 **B**「コマンド行によるアンインストール、インストール、および設定」を参照してください。**Performance Reporting Manager** などのほかのアドオンソフトウェアを同時にインストールすることもできます。

▼ Sun Management Center 3.6 へのアップグレード時に Container Manager をインストールする

手順

- 詳細および具体的な手順については、『**Sun Management Center 3.6** インストールと構成ガイド』の第 **5** 章「旧バージョンの **SyMON** および **Sun Management Center** のアップグレード **(Solaris** プラットフォーム**)**」を参照してください。**Performance Reporting Manager** などのほかのアドオンソフトウェアを同時にインストールすることもできます。

▼ Container Manager を別個にインストールする

手順 **1.** スーパーユーザー **(su -)** で、次のように入力して **Sun Management Center 3.6** インストールスクリプトを実行します。

```
# /opt/SUNWsymon/sbin/es-inst
```

/opt は、Sun Management Center 3.6 がインストールされているディレクトリを示します。これとは異なるディレクトリの場合には、実際の名前に置き換えてください。

2. プロンプトが表示されたら、**Container Manager** ファイルのソースディレクトリを指定します。

```
Please enter the source directory:
```

- SPARC システムでは、次のいずれかを選択します。
  - ソフトウェア CD からインストールする場合は、次のように入力します。

    ```
    # /<DiskMountDir>/image
    ```
  - ソフトウェアがコピーされているディレクトリからインストールする場合は、次のように入力します。

    ```
    # disk1/image
    ```

    disk1 は、ソフトウェアがコピーされた場所です。
- x86 システムでは、次のいずれかを選択します。
  - ソフトウェアディスクからインストールする場合は、次のように入力します。

    ```
    # /<DiskMountDir>/x86/image
    ```
  - ソフトウェアがコピーされているディレクトリからインストールする場合は、次のように入力します。

    ```
    # disk1/x86/image
    ```

    disk1 は、ソフトウェアがコピーされた場所です。

3. **Container Manager** ソフトウェアをインストールするには、プロンプトに対して「**Yes**」と応答します。

es-inst スクリプトによって Container Manager がインストールされます。次に、自動的に setup プロンプトが表示されます。

4. この時点で **Container Manager** の設定を行うか、あとで行うかを決めます。

次の例に示すような出力が表示されます。

```
You must perform setup before using Sun Management Center 3.6.
Do you want to run setup now (y|n|q)
```

- あとで行う場合は、「No」を表す n で応答します。設定処理を実行する準備ができたら、39 ページの「Container Manager を別個に設定する」を参照してください。
- この時点で設定を行う場合は、「Yes」を表す y で応答します。

Container Manager の設定

インストールが完了したあと、Container Manager の `setup` スクリプトを実行して、サーバーレイヤとエージェントレイヤの設定を行う必要があります。ソフトウェアのインストールと同様に、Sun Management Center 3.6 のインストールまたはアップグレードを設定するときに設定処理を実行できます。ソフトウェアをあとで別個に設定することもできます。

コマンド行から設定処理を実行した場合に設定ファイルに対して行われる変更の内容は、設定ウィザードを使用した場合の変更内容と同じです。詳細は、37 ページの「Container Manager ソフトウェアの設定」を参照してください。

▼ Sun Management Center のインストールまたはアップグレード時に Container Manager を設定する

手順

- 詳細および具体的な手順については、『**Sun Management Center 3.6** インストールと構成ガイド』の付録 **B**「コマンド行によるアンインストール、インストール、および設定」を参照してください。

▼ Container Manager を別個に設定する

注 – この手順では、インストール処理の最後に Container Manager の設定を行わないことを選択していることを前提とします。

手順

1. スーパーユーザー **(su –)** で、**Sun Management Center 3.6 `setup`** スクリプトを実行します。次のいずれか **1** つを指定してください。
   - まだ設定を行なっていないすべてのコンポーネントの設定を実行するには、次のように入力します。

```
# /opt/SUNWsymon/sbin/es-setup
```

- Container Manager アドオンソフトウェアの設定のみを実行するには、次のように入力します。

```
# /opt/SUNWsymon/sbin/es-setup -p SCM
```

/opt は、Sun Management Center 3.6 がインストールされているディレクトリを示します。これとは異なるディレクトリの場合には、実際の名前に置き換えてください。

2. 設定処理を継続するために停止する必要があるコンポーネントのリストを確認します。

3. **Container Manager** が、設定する製品であることを確認します。

4. エージェントレイヤを設定するとき、リソースプールに関連付けられていないシステムでプロセッサセットが検出された場合は、次のいずれかを選択します。

   - 「はい」– このプロセッサセットはシステムから削除されます。Container Manager が正常に機能するには、リソースプールに関連付けられていないプロセッサセットは、プール管理の問題の原因になる可能性があるので、削除する必要があります。このオプションを選択したら、プロセッサセットは自動的に削除され、設定処理が続行します。
   - 「いいえ」– このプロセッサセットはシステムから削除されません。このオプションを選択したら、設定スクリプトが終了し、Container Manager ソフトウェアはシステムで設定されません。リソースプールに関連付けられていないプロセッサセットがあるホスト上では、Container Manager ソフトウェアを使用できません。
   - 「取消し」– 設定スクリプトが終了します。Container Manager ソフトウェアはシステムで設定されません。

5. コンマ区切りで有効なユーザー名を入力します。ここで入力するユーザーは、サーバーマシン上で **/var/opt/SUNWsymon/cfg/esusers** ファイルにすでに存在し、かつ **esadm** または **esdomadm** グループに割り当てられている必要があります。

   ユーザーおよびプロファイルの両方が正しく存在する場合は、必要なプロファイル (プロジェクト管理、プール管理、ゾーン管理) に、これらのユーザーが追加されます。プロファイルは、Solaris のバージョンに基づきます。

   各エージェントマシンでは、/opt/SUNWsymon/addons/SCM/sbin/scm-poolprof-script.sh スクリプトを実行してプロファイルを作成します。/opt は、Container Manager がインストールされているディレクトリです。このスクリプトを実行するにはJAVA_HOME 環境変数が、/usr/j2se に設定されているか、Java が /usr/java にインストールされている必要があります。

   このスクリプトを実行しても、Solaris 9 または 10 のゾーン管理プロファイル、プロジェクト管理プロファイルは作成されません。このスクリプトでは、Solaris 9 または 10 のプール管理プロファイルまたは Solaris 8 のプロジェクト管理プロファイルだけが作成されます。

注意 – 事前定義された Solaris プロファイルを 1 つでも削除した場合、Container Manager は正常に動作しない可能性があります。

注 – この手順で追加されたユーザーだけが、Solaris 10 のゾーン、Solaris 9 および 10 のリソースプール、Solaris 8、9、10 のプロジェクトを管理することができます。

6. ほかの **Sun Management Center** 製品の設定処理を以前に行った場合は、すべてのソフトウェアの設定を再度、実行するか、最新のアドオンソフトウェアだけの設定を実行するかを指定します。

7. サーバーレイヤのインストールを設定する場合は、データベース用に **300M** バイト以上の空きディスク容量があるディレクトリのフルパスを指定します。

8. **Y** または **N** を入力して、**Sun Management Center** のエージェントとサーバーのコンポーネントを起動するかどうかを指定します。

   ソフトウェアが設定されたら、確認のメッセージが表示されます。

   設定手順を完了すると、Sun Management Center のプロセスを開始できます。詳細は、『Sun Management Center 3.6 インストールと構成ガイド』の第 8 章「Sun Management Center の起動と停止」を参照してください。

例 **A–1** 設定完了後の出力例

設定が完了し、Sun Management Center のエージェントレイヤとサーバーレイヤが起動したあとに表示される出力の例を次に示します。

```
Do you want to start Sun Management Center agent and server components now (y|n|q) y

Java server started successfully.
Grouping service started successfully.
Agent started successfully.
Topology service started successfully.
Trap-handler service started successfully.
Configuration service started successfully.
Event-handler service started successfully.
Metadata Service started successfully.
Hardware service started successfully.


Web server started successfully.

Setup of Sun Management Center is complete.
Setup logfile is : /var/opt/SUNWsymon/install/setup_host_name.040408141450.12822

Install logfile is : /var/opt/SUNWsymon/install/install_host_name.040408140547.10929
```

```
                    End of Installation

Exiting Sun Management Center installation.

#
```

Container Manager ソフトウェアの削除

`es-uninst` スクリプトによって Container Manager アドオンソフトウェアが削除されます。このスクリプトでは、Container Manager のパッケージと、設定時に行われたデータと設定の変更がすべて削除されます。ソフトウェアパッケージに加えて、`base-modules-d.dat` のモジュールエントリと Java Web Console のアプリケーション登録が削除されます。削除処理中には、データベースに保存されているデータを残すかどうかを確認するメッセージが表示されます。インストール時に行われたシステムリソースの設定の変更は、Container Manager ソフトウェアのアンインストール時には削除されません。

▼ `es-uninst` を使用して Container Manager を削除する

手順 **1.** スーパーユーザー **(su –)** で、次のように入力します。

```
# /opt/SUNWsymon/sbin/es-uninst
```

`/opt` は、Sun Management Center 3.6 がインストールされているディレクトリを示します。これとは異なるディレクトリの場合には、実際の名前に置き換えてください。

2. アドオンソフトウェアのリストから **Container Manager** を選択します。

3. データファイルを保存するかどうかを指定します。

表示される出力の例を次に示します。

```
Select Save Data to save all user and configuration data. Your data is
saved and can be restored when you re-install Sun Management Center.
Do you want to preserve data (y|n|q)
```

`es-uninst` スクリプトが完了すると、Container Manager のパッケージと設定ファイルが削除されます。データファイルの保存を指定しない場合は、データファイルも削除されます。

用語集

IPQoS	IPQoS (IP Quality of Service) 機能によって、Solaris ソーンのインバウンドトラフィックおよびアウトバウンドトラフィックを制御することで、ネットワークユーザーへのサービスのレベルを一定に保つことができます。
一致式	アプリケーションに関連付けられているプロセスを特定するための式。
関連付け	プロジェクトをホストにバインドすること。
共有メモリー	プロジェクト内で実行されるプロセスに使用できるメモリーの総容量。
公平配分スケジューラ **(FSS)**	シェア数を基準に CPU 時間を割り当てるスケジューリングクラス。シェアは、システムの CPU リソースのうちプロジェクトに割り当てる部分を定義します。
コンテナ	後で使うために作成し、名前を付け、保存したプロジェクト。この場合のプロジェクトは、ホストに関連付けられておらず、またリソース境界が設定されていません。この製品の 1.0 バージョンでは、コンテナのことを「コンテナ定義」と呼んでいました。
制限ノード **(lnode)**	Solaris 8 で、ユーザーごとのデータが保持される機構。
ゾーン	Solaris 10 オペレーティングシステムが動作しているシステムで設定できる仮想化オペレーティングシステム環境。すべての Solaris システムには、システムのデフォルトのゾーンである大域ゾーンが含まれます。非大域ゾーンは、作成、削除、変更、停止、および再起動できます。
タイムシェアスケジューラ **(TS)**	CPU 時間を優先順位に従って割り当てることで、使用可能な CPU へのアクセスを比較的、均等にすべてのプロセスに配分しようとするスケジューリングクラス。
動的リソースプール	Solaris 10 の機能である動的リソースプールによって、システムイベントとアプリケーションの負荷の変化に応じて、各プールのリソース割り当てを調整できます。

配備済み	「有効なプロジェクト」を参照。
プール	「リソースプール」を参照。
プロジェクト	ホストに関連付けられているコンテナ。一連の物理システムリソースの構成と管理に役立つ抽象層です。 Solaris 9 では、プロジェクトは、関連する作業のネットワーク全体での管理識別子になります。
プロジェクト識別子 **(ID)**	Solaris カーネルでリソースの使用状況が追跡される手段。コンテナ内で実行されるプロセスは、すべてプロジェクト ID が同じです。
ホスト	ホストとは、Container Manager のエージェントソフトウェアがインストールされ、また Sun Management Center サーバーのコンテキストに含まれるシステムです。インストールが完了すると、ホストは自動的に検出され、その名前が「ホスト」表示のナビゲーションウィンドウに追加されます。
無効なプロジェクト	ホストに関連付け、リソース境界を設定したが、リソース境界が現在はカーネルによって適用されていないプロジェクト。
有効なプロジェクト	ホストに関連付け、最小 CPU 予約数とメモリーキャップを設定したプロジェクト。有効なプロジェクトは、関連付けられているホストの `/etc/project` ファイルにエントリがあります。有効なプロジェクトのリソース境界は、カーネルによって適用されます。有効なプロジェクトとは、外に出されてホストで有効になっているということから、配備済みともいえます。
リソース	リソース管理において、アプリケーションの動作を変更するために操作可能な計算機システムの一部。
リソース管理	利用可能なシステムリソースのアプリケーションによる使用方法を制御可能にする機能。
リソースプール	リソース管理において、マシンのリソースの分割に使用される構成機構。「プール」ともいいます。

索引

C

Container Manager GUI
  起動，50
  タブの概要，52-54
Container Manager のアンインストール
  ウィザードの使用，42-43
  コマンド行の使用，139
Container Manager のインストール
  オプション，35
  コマンド行の概要，133-136
Container Manager の使用例，26-27
Container Manager の設定
  GUI の概要，37
  概要, コマンド行，136-139
  設定の変更，37
Container Manager をインストールする前に，25
CPU
  CPU リソースプールで使用可能，104
  最小予約，18-19
CPU シェア数, 概要，62-71
CPU 予約数, シェア数との関係，63
CSV レポート，125
CSV レポートのエクスポート，125

E

`es-inst` スクリプト，133
`es-uninst` スクリプト，139
`/etc/pooladm.conf`, 変更，38
`/etc/project` ファイル
  コンテナの無効化，50
`/etc/project` ファイル (続き)
  デフォルトのコンテナ，59
  プロジェクトの有効化，50
  プロジェクト名，61

F

FSS (公平配分スケジューラ), CPU 予約数，63

G

`group.staff` コンテナ，59

I

IPQoS，28
IP Quality of Service，28

J

Java Web Console，51

L

`lnode`, 定義，21

M

MIB 表，104

P

Performance Reporting Manager
　データ収集サービス，126
　レポートの要件，24
pool_default，22, 103

S

Solaris 8, リソースプール，22
Solaris Container モデル、概要，19
Solaris OS のバージョン, 製品の機能，34
Solaris OS バージョン, プロジェクトの種類，61
Solaris Container Manager 3.6 の機能，27-29
Solaris Container Manager 3.6 の新機能，27-29
Sun Management Center コンソール，34
Sun Management Center と Solaris Container Manager
　RAM の要件，32-34
　オペレーティングシステムの要件，32-34
　スワップ空間の要件，32-34
　ディスク容量の要件，32-34
Sun Web Console, 登録解除，42-43

あ

アプリケーションベースのプロジェクト, プロセスを個別に移動，86
アプリケーションベースプロジェクト
　概要，61, 80-85
アラームしきい値
　概要，119-121
　削除，121
　設定，120-121
アラームしきい値の削除，121
アラームしきい値の設定，120-121
アラームの管理, 前提条件，119-121

い

一致式
　使用しないでプロセスを移動，85-88
　説明，50
　プロセスの自動移動，81
「移動」ボタン、使用可能にする，59
インストール
　Performance Reporting Manager，35
　Sun Management Center のレイヤ，31
　概要，31
　製品マニュアル，36

う

ウィザード
　新規プロジェクト，74
　ホストのコンテナへの関連付け，88

え

エージェント更新，42
エージェントの検出，55

か

カーネル, 適用，45
拡張アカウンティングデータ，24
拡張アカウンティングファイル, 変更，38
関連マニュアル，29-30

き

起動, Container Manager GUI，50
共有メモリー，79

く

グラフレポート
　Performance Reporting Manager 要件，24
　可用性，123
　種類，124
グラフレポートの種類，124
グループ、使用，58-59

グループベースプロジェクト, 概要，61

け
検出、エージェント，55

こ
コンテナ
  group.staff，59
  インスタンス，48
  機能，45
  デフォルト，59
  特長，46
  とプロジェクト，49
  プロパティー，47
  変更，95
  有効，50
  リソース使用状況，45
  リソースの制御，18-19
コンテナ管理, 概要，45
コンテナの境界, 作成，22
コンテナの作成, 概要，61-71
「コンテナ」表，55
コンテナ表示, 概要，57
コンテナまたはホストの移動，58-59

さ
サーバーコンテキスト，42, 55
サーバー統合計画
  アプリケーションリソースの使用状況の傾向，63
  作業，25
サービス
  定義，19
  リソースとの関係，19
作成, プロファイル，41

し
柔軟性のあるプロセス管理，28
状態、プロジェクト，49
新規プロジェクトウィザード, 事前に必要な情報，74

せ
製品マニュアルマニュアル, インストール後に表示，36
設定時の設定変更，37

そ
ゾーン
  概要，109-110
  状態，110
  説明，23-24
  ログファイルの表示，118
ゾーンのログファイルの表示，118

た
大域ゾーン, 説明，23
タイムシェアスケジューラ，69-70
  概要，29
タブ
  「アラームしきい値」，52
  「コンテナ」，52
  「コンテンツ」，52
  「使用状況」，52
  「ゾーン」，52
  「プロセス」，52
  「プロパティ」，52
  「ホスト」，52
  リソース変更ジョブ，52

ち
注
  Solaris 8 OS で使用可能なコンテナの種類，77
  Solaris 8 OS でのデフォルトのコンテナ，59
  追加ソフトウェアのインストール，32
  変更前のコンテナの無効化，95
  モジュールの読み込み，34
  リソース予約の変更，98

注意
CPU シェア数の手動変更，68
Sun Management Center サーバーコンテキスト，55
メモリーキャップ不足，63
リソース管理コマンドの使用，38

て
データ収集プロセス，126
デフォルトのコンテナ
Solaris 8 OS での一覧，59
Solaris 9 OS での一覧，59
概要，59-60
リソース予約，60
デフォルトプール，103-104
デフォルトプロジェクト，60

と
動的リソースプール，28

は
ハードウェアプラットフォーム, 一覧，34
配備済みプロジェクト，50
バックアップと復元，41
パッケージ, 一覧，34

ひ
非大域ゾーン
起動，117-118
コピー，116-117
削除，117-118
作成，110
説明，24
停止，117-118
非大域ゾーンの起動，117-118
非大域ゾーンのコピー，116-117
非大域ゾーンの削除，117-118
非大域ゾーンの作成，110
非大域ゾーンの停止，117-118

表
MIB，104
コンテナ，55
プロセス，91-93
リソースプール，104
リソース変更ジョブ，98

ふ
負荷, 環境のカスタマイズ，18-19
ブラウザ、サポート対象の一覧，50
プロジェクト
アプリケーションベース，80-85
削除，102
作成，73-85
作成に必要な情報，74
種類との関連，61
状態，49
定義，21
デフォルト，60
配備済み，50
プロセスの表示，91-93
プロセスを個別に移動，86
プロパティシートで変更，93-101
無効化，50
有効化または無効化，88-91
ユーザーベースまたはグループベース, 作成，77-80
リソースの追跡，45, 61
リソース変更ジョブ，97
リソース変更ジョブで変更，97-101
プロジェクト ID
カーネルによる追跡，21
プロセスの追跡，61
プロジェクト内のプロセスの開始, 概要，85-88
プロジェクトの削除，102
プロジェクトの作成，73-85
プロジェクトの手動での移動, 概要，85-88
プロジェクトの種類，61-62
識別子の指定，78
詳細，61
プロジェクトの無効化，88
概要，50
プロジェクトの有効化，88
失敗，61
プロジェクトの有効化の失敗，61
「プロジェクト」表，55

プロジェクト名
　重複，61
　種類に基づく，61
プロセス
　一致式で移動，81
　個別に移動，86
　手動で移動，81, 85-88
　デフォルトコンテナ，60
　プロジェクト ID，61
　プロジェクト内で開始，81, 85-88
「プロセス」表
　フィールドの説明，91-93
　プロセスが表示されない，92
プロセッサセット, 既存，40
プロパティシート
　相違，93-101
　フィールドの説明，96, 97
　リソースプールまたはリソース予約の変更，94
プロファイルの作成，41

ほ
ホストに関連付け，50
ホスト表示, 概要，54-57

み
未確認のアラーム, 表示，121
未確認のアラームの表示，121

め
メモリーキャップ, 不十分なレベルとパフォーマンス，62
メモリー予約，18-19

も
モジュールの読み込み，34

ゆ
有効なプロジェクト
　アプリケーションベース，80-85
　概要，50
　重複するプロジェクト名，61
　プロセスの表示，91-93
ユーザーベースプロジェクト, 概要，61

り
リソース, 定義，21
リソース管理, 概要，21
リソース管理コマンド、使用，50
リソース競合, 制限，18-19
リソース使用状況
　制御，45
　追跡，45, 46
　レポート，123-125
リソース使用状況の傾向，62
リソース使用状況レポート
　コンテナ，129
　ゾーン，131-132
　ホスト，127-128
　有効なプロジェクト，128
　リソースプール，129-131
リソース消費, ニーズの評価，25
リソースセット, 定義，22
リソースの制御，23
リソースの予約, システムリソースの均衡化，45
リソースプール
　Solaris 8，103-104
　既存，103
　コンテナ数，22
　コンテナとの関係，22
　削除，107
　作成の概要，104
　定義，22
　ナビゲーションウィンドウでの位置，103
　プロジェクト作成時に新規プールを作成，84
　プロパティシートで変更，94
　変更，106
　リソースの分割，103-104
リソースプールの削除，107
リソースプールの変更，106
「リソースプール」表，104
リソース変更ジョブ
　Sun Management Center のジョブ，99

リソース変更ジョブ (続き)
概要，97-101
システムリソースの管理に使用，98
リソース変更ジョブのスケジュール，98
「リソース変更ジョブ」表, フィールド，98
リソース予約
カーネルによる適用，49
概要，62-71
初期設定，63
プロパティシートで変更，94
メモリー不足とパフォーマンス，62

れ

レポート
CSV にエクスポート，125
要求，126-132