

Oracle® Solaris 10 JumpStart から Oracle  
Solaris 11.2 自動インストーラへの移行

ORACLE®

Part No: E53766  
2014 年 7 月

Copyright © 2011, 2014, Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

このソフトウェアもしくはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアもしくはハードウェアは、危険が伴うアプリケーション（人的傷害を発生させる可能性があるアプリケーションを含む）への用途を目的として開発されていません。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用する場合、安全に使用するために、適切な安全装置、バックアップ、冗長性（redundancy）、その他の対策を講じることは使用者の責任となります。このソフトウェアもしくはハードウェアを危険が伴うアプリケーションで使用したことに起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはOracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

Intel, Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD, Opteron, AMDロゴ, AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

# 目次

---

このドキュメントの使用方法 .....	7
<b>1 JumpStart と自動インストーラの比較 .....</b>	<b>9</b>
JumpStart と AI の類似点と相違点 .....	9
JumpStart から AI への移行情報 .....	10
<b>2 JumpStart ルールおよびプロファイルファイルの変換 .....</b>	<b>13</b>
AI クライアントの条件および AI マニフェストについて .....	13
ルールのキーワードと条件の指令の比較 .....	13
プロファイルのキーワードと AI マニフェストの指令の比較 .....	15
js2ai の使用による JumpStart ルールおよびプロファイルから AI 条件およびマニ フェストへの変換 .....	22
js2ai による JumpStart デバイス指定の変換方法 .....	23
ソフトウェアパッケージの変換 .....	25
js2ai を使用したルールおよびプロファイルの変換 .....	28
追加のプロファイル変換情報の表示 .....	37
出力 AI マニフェストの検証 .....	38
<b>3 sysidcfg 構成ファイルの変換 .....</b>	<b>39</b>
sysidcfg ファイルキーワードとシステム構成プロファイル指令の比較 .....	39
js2ai を使用した sysidcfg ファイルからシステム構成プロファイルへの変換 .....	41
js2ai 変換の警告 .....	42
js2ai を使用した sysidcfg の変換 .....	43
詳細な構成変換情報の表示 .....	45
出力システム構成プロファイルの検証 .....	46
<b>4 Oracle Solaris 11 サーバー上の JumpStart を使用した Oracle Solaris 10 のイン     ストール .....</b>	<b>47</b>
Oracle Solaris 10 JumpStart サーバーとしての Oracle Solaris 11 システムの 設定 .....	47

▼ Oracle Solaris 10 JumpStart サーバーとして Oracle Solaris 11 システムを設定する方法 .....	47
索引 .....	53

## 表目次

---

表 1-1	JumpStart と AI のタスクの比較 .....	9
表 2-1	JumpStart ルールファイルのキーワードと AI 条件の指令の比較 .....	14
表 2-2	JumpStart プロファイルファイルのキーワードと AI マニフェストの指令の 比較 .....	16
表 3-1	sysidcfg ファイルキーワードと構成プロファイル指令の比較 .....	39



## このドキュメントの使用方法

---

- 概要 – Oracle Solaris 10 JumpStart から Oracle Solaris 11 Automated Installer に移行する方法を示します。
- 対象読者 – システム管理者
- 必要な知識 – ある程度 Oracle Solaris の経験があれば役に立ちます。

## 製品ドキュメントライブラリ

この製品の最新情報や既知の問題は、ドキュメントライブラリ (<http://www.oracle.com/pls/topic/lookup?ctx=E56342>) に含まれています。

## Oracle サポートへのアクセス

Oracle のお客様は、My Oracle Support を通じて電子的なサポートを利用することができます。詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> を参照してください。聴覚に障害をお持ちの場合は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> を参照してください。

## フィードバック

このドキュメントに関するフィードバックを <http://www.oracle.com/goto/docfeedback> からお聞かせください。



# ◆◆◆ 第 1 章

## JumpStart と自動インストーラの比較

---

この章では、次の内容について説明します。

- JumpStart と自動インストーラ (AI) の類似点と相違点
- JumpStart から AI への移行に役立つ参照情報

### JumpStart と AI の類似点と相違点

JumpStart と AI の両方とも、手入力なしでネットワーク上の複数のシステムのインストールを行うことができます。クライアントはネットワークブートされ、クライアントがブートすると、インストーラが引き継ぎます。

JumpStart は、Oracle Solaris 10 OS およびそれ以前の Oracle Solaris OS のバージョンをインストールします。AI は Oracle Solaris 11 OS と更新リリースをインストールします。

JumpStart と AI には、次の共通した特徴があります。

- システム構成をインストールサーバーに格納することで、手入力なしで複数のクライアントのネットワークインストールを行います。
- 1 回の自動インストールで、異なるクライアントに対して異なる種類のインストールを行います。
- x86 と SPARC の両方のクライアントをインストールします。

表 1-1 JumpStart と AI のタスクの比較

段階	JumpStart	AI
インストールサーバーを設定します。	<code>setup_install_server</code> コマンドを使用します。	<code>installadm create-service</code> コマンドを使用します。
インストールにクライアントを追加します。	<code>add_install_client</code> コマンドを使用します。	<code>installadm create-client</code> コマンドを使用します。

段階	JumpStart	AI
派生したプロファイルを作成します。	開始スクリプトを使用します。	派生したマニフェストメカニズムを使用します。
インストール手順を指定します。	プロファイルファイルを使用します。	AI マニフェストファイルを使用します。
クライアントをプロビジョニングします。	DVD を使用します。	インターネットまたはローカルネットワーク上のパッケージリポジトリを使用します。
クライアントのカスタマイズを指定します。	ルールファイルを使用して、クライアントをプロファイルファイルに関連付けます。	<code>installadm</code> コマンドの <code>create-manifest</code> 、 <code>create-profile</code> 、または <code>set-criteria</code> サブコマンドを使用して、クライアントを AI マニフェストおよびシステム構成プロファイルに関連付けます。
クライアント構成を指定します。	終了スクリプトと <code>sysidcfg</code> ファイルを使用します。	SMF (Oracle Solaris Service Management Facility) システム構成プロファイルファイルを使用します。初回ブート時に 1 回実行される SMF サービスによって実行されるスクリプトを使用します。

## JumpStart から AI への移行情報

次の各項目に、Automated Installer の使用に移行する際に有用ないくつかの戦略を示します。

- **JumpStart** ルール、プロファイル、および構成ファイルを **AI** 条件、**AI** マニフェスト、および **SMF** システム構成プロファイルに変換します。

`js2ai` コマンドを取得します。

```
# pkg install install/js2ai
```

JumpStart データを AI データに変換するには、次のページの手順に従ってください。

- [js2ai\(1M\)](#) のマニュアルページ
- [第2章「JumpStart ルールおよびプロファイルファイルの変換」](#)
- [第3章「sysidcfg 構成ファイルの変換」](#)
- 1 つのサーバーを **JumpStart** インストールサーバーおよび **AI** インストールサーバーの両方として使用します。

この構成を使用すると、JumpStart 対応のシステムを Oracle Solaris 11 サーバーからサポートできます。詳細は、[第4章「Oracle Solaris 11 サーバー上の JumpStart を使用した Oracle Solaris 10 のインストール」](#)を参照してください。

- **AI クライアントプロビジョニングマニフェストを動的に派生させます。**

『Oracle Solaris 11.2 システムのインストール』の「[クライアントインストール時の AI マニフェストの作成](#)」を参照してください。JumpStart 開始スクリプトは、インストーラに渡されるインストールパラメータを動的に操作する機能を提供します。AI は、クライアントのインストール時にクライアント属性を照会し、そのクライアント向けにカスタマイズされたプロビジョニングマニフェストを動的に派生させることができます。環境変数はクライアントのハードウェア属性を指定し、これらの属性のほとんどは、JumpStart 開始スクリプトで使用される環境変数と同じです。

- **AI インストール用のソフトウェアパッケージリポジトリにアクセスします。**

インターネット上にある Oracle Solaris 11 パッケージリポジトリを使用してください ([pkg.oracle.com](http://pkg.oracle.com) など)。

パッケージリポジトリのローカルコピーを作成します (『Oracle Solaris 11.2 パッケージリポジトリのコピーと作成』)。

- **システム構成の指示を行います。**

SMF プロファイルの作成については、『Oracle Solaris 11.2 システムのインストール』の [第 11 章「クライアントシステムの構成」](#)を参照してください。

AI クライアントインストールの一部として非大域ゾーンをインストールするために AI マニフェストおよび SMF プロファイルを作成することについては、『Oracle Solaris 11.2 システムのインストール』の [第 12 章「ゾーンのインストールと構成」](#)を参照してください。

- **初回ブート時に 1 回実行してユーザー定義スクリプトを実行する SMF サービスを作成します。**

『Oracle Solaris 11.2 システムのインストール』の [第 13 章「初回ブート時のカスタムスクリプトの実行」](#)を参照してください。

- **あるいは、カスタムの ISO イメージを作成して、必要なインストールを作成します。**

カスタムインストールイメージの作成については、『Oracle Solaris 11.2 カスタムインストールイメージの作成』を参照してください。



## ◆◆◆ 第 2 章

# JumpStart ルールおよびプロファイルファイルの変換

---

この章では、JumpStart ルールおよびプロファイルファイルを AI 条件ファイルおよび AI マニフェストに変換する `js2ai` コマンドの使用方法を示します。

## AI クライアントの条件および AI マニフェストについて

AI マニフェストはディスクレイアウトやインストールするソフトウェアパッケージなどのシステムプロビジョニングを指定する XML ファイルです。AI マニフェストについては、『[Oracle Solaris 11.2 システムのインストール](#)』の第 10 章「クライアントシステムのプロビジョニング」および [ai\\_manifest\(4\)](#) のマニュアルページを参照してください。

AI ではクライアント条件を使用して、クライアントシステムがインストールを完了するために使用する AI マニフェストファイルを指定します。AI マニフェストが AI インストールサービスに追加されるとき、コマンド行またはファイルで条件を指定できます。AI クライアント条件の指定の詳細は、『[Oracle Solaris 11.2 システムのインストール](#)』の第 9 章「インストールのカスタマイズ」を参照してください。

## ルールのキーワードと条件の指令の比較

次の表では、JumpStart ルールのキーワードと AI 条件の指令を比較します。AI はこれらの条件を使用して、正しい AI マニフェストまたはシステム構成プロファイルを特定のクライアントに適用します。

表 2-1 JumpStart ルールファイルのキーワードと AI 条件の指令の比較

JumpStart ルールファイル のキーワード	AI 条件ファイルの指令	コマンド行の例	条件ファイルの例
any	サポートされていません。 どの選択条件にも一致し ないクライアントシステム の場合、AI インストール サービスはデフォルトの AI マニフェストを提供し ます。		
arch	cpu	-c cpu=sparc	<ai_criteria name="cpu"> <value>sparc</value> </ai_criteria>
disksize	サポートされていませ ん。AI マニフェストで指 定されたターゲットのディ スクが最低限必要なサイ ズより大きい場合、AI は ターゲット上にインストー ルします。		
domainname	サポートされていません。		
hostaddress	ipv4	-c ipv4=10.6.68.127	<ai_criteria name="ipv4"> <value>10.6.68.127</value> </ai_criteria>
hostname	サポートされていま せん。AI でホストを 一意に識別するに は、hostaddress で説明 した IP アドレスまたは MAC アドレスを使用し ます。	-c mac=0:3:ba:33:9d:b6	<ai_criteria name="mac"> <value>0:3:ba:33:9d:b6</ value> </ai_criteria>
installed	サポートされていませ ん。AI マニフェストで指 定されたターゲットのディ スクが最低限必要なサイ ズより大きい場合、AI は ターゲット上にインストー ルします。		
karch	arch	-c arch=i86pc	<ai_criteria name="arch"> <value>i86pc</value> </ai_criteria>
memsize	mem	-c mem=2048	<ai_criteria name="mem"> <value>2048</value>

JumpStart ルールファイル のキーワード	AI 条件ファイルの指令	コマンド行の例	条件ファイルの例
			</ai_criteria>
model	platform	-c platform=SUNW,Sun-Fire-T200	<ai_criteria_name="platform"> <value>SUNW,Sun-Fire-T200</value> </ai_criteria>
network	network  network、範囲を指定した network、または範囲を指定した ipv4 を使用します。	単一の network 値:  -c network="10.0.0.0"	単一の network 値:  <ai_criteria name="network"> <value>10.0.0.0</value> </ai_criteria>
	ipv4	ipv4 範囲:  -c ipv4=10.0.0.1-10.0.0.64	ipv4 範囲:  <ai_criteria name="ipv4"> <range>10.0.0.1 10.0.0.64</range> </ai_criteria>
osname	サポートされていません。		
probe	サポートされていません。		
totaldisk	サポートされていません。AI マニフェストで指定されたターゲットのディスクが最低限必要なサイズより大きい場合、AI はターゲット上にインストールします。		

## プロファイルのキーワードと AI マニフェストの指令の比較

次の表では、JumpStart プロファイルのキーワードと AI マニフェストの指令を比較します。AI は、XML マニフェストファイルを使用してクライアントインストールを定義します。AI マニフェストについては、『Oracle Solaris 11.2 システムのインストール』の第 10 章「クライアントシステムのプロビジョニング」および [ai\\_manifest\(4\)](#) のマニュアルページを参照してください。

any で指定されるデバイスなど、クライアントのインストールプロセスが開始するまで不明な値を指定する場合、派生したマニフェストスクリプトの使用を検討してください。派生したマニフェストスクリプトを使用して、たとえばディスクサイズに基づいてスワップサイズを指定したり、使用可能なディスクに基づいてミラー化を指定したりすることができます。派生したマニフェスト

スクリプトについては、『Oracle Solaris 11.2 システムのインストール』の「クライアントインストール時の AI マニフェストの作成」を参照してください。

表 2-2 JumpStart プロファイルファイルのキーワードと AI マニフェストの指令の比較

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
archive_location	サポートされていません。AI はフラッシュアーカイブからインストールしません。
backup_media	サポートされていません。backup_media キーワードは install_type の upgrade オプションと一緒にのみ使用します。AI はアップグレードインストールタイプをサポートせず、初期インストールのみサポートします。
boot_device device  例:  boot_device c1t0d0	例:  <target> <disk whole_disk="true"> <disk_name name="c1t0d0" name_type="ctd"/> </disk> <logical nodump="true" noswap="false"/> </target>
boot_device device eeprom  例:  boot_device c0t0d0s0 update	eeeprom キーワードの値 (SPARC システムの場合は update、x86 システムの場合は preserve) は、AI ではサポートされていません。AI では、SPARC システムの EEPROM は、インストールしたシステムが指定したターゲットデバイスから自動的にブートするように、常にそのデバイスに更新されます。x86 システムでは、ファームウェアは更新されません。
bootenv	サポートされていません。AI はブート環境を作成し、Oracle Solaris 11 OS をそのブート環境にインストールします。
client_arch	サポートされていません。インストールされるクライアントアーキテクチャーは AI マニフェストでなく AI インストールサービスによって定義されます。
client_root	サポートされていません。ターゲットディスクの種類、ベンダー、およびサイズを指定できます。ターゲットディスクのルート領域の量は指定できません。ai_manifest(4) のマニュアルページを参照してください。
client_swap	サポートされていません。デフォルトでは、AI はルートプールにスワップボリュームを作成します。スワップスライスを指定できます。ai_manifest(4) のマニュアルページを参照してください。
cluster cluster-name	サポートされていません。cluster および package 指定の扱い方については、25 ページの「ソフトウェアパッケージの変換」を参照してください。

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
	<p>Oracle Solaris 11 OS は、<a href="#">pkg(5)</a>のマニュアルページに定義されているグループパッケージを使用します。グループパッケージは、マニフェストでほかのパッケージを指定するようにして指定します。デフォルトの AI マニフェストには、標準の Oracle Solaris 11 インストールに必要なパッケージが含まれています。このパッケージリストはカスタマイズ可能です。</p>
<p><code>cluster cluster-name delete</code></p>	<p>サポートされていません。<code>delete</code> スイッチは、<code>install_type</code> の <code>upgrade</code> オプションと一緒にのみ使用します。AI はアップグレードインストールタイプをサポートせず、初期インストールのみサポートします。</p>
<p><code>dontuse</code></p>	<p>サポートされていません。</p>
<p><code>fdisk disk-name type size</code></p> <p>例:</p> <p><code>fdisk c0t3d0 solaris all</code></p>	<p>例:</p> <pre data-bbox="824 831 1383 1041">&lt;target&gt;   &lt;disk&gt;     &lt;disk_name name="c0t3d0" name_type="ctd"/&gt;     &lt;partition action="create" name="1" part_type="191"/&gt;   &lt;/disk&gt;   &lt;logical nodump="true" noswap="false"/&gt; &lt;/target&gt;</pre> <p>AI でサポートされているディスクおよびパーティション属性の完全な一覧については、<a href="#">ai_manifest(4)</a>のマニュアルページを参照してください。</p> <p>js2ai による変換:</p> <p>js2ai 変換の場合、<code>disk_name</code> の値はデバイスにする必要があります。all のデバイスはサポートされていません。fdisk タイプは solaris にする必要があります。サイズ 0 または delete はサポートされていません。パーティション分割が default で、rootdisk が設定されていない場合、js2ai は最初に見つかった fdisk solaris パーティションをルートディスクとして設定します。</p>
<p><code>filesys</code></p>	<p>UFS ファイルシステムはサポートされません。AI は ZFS ファイルシステムをインストールします。</p> <p>js2ai による変換:</p> <p>ルートに使用するデバイスを決定するための方法が他にない場合、filesys の行で / のマウントポイントが付いたデバイスがルートプールとして使用されます。</p> <p>指定されたマウントポイントが / または swap の場合は、ローカルおよびミラー化されたファイルシステムがサポートされます。</p>

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
	<p>マウントポイントが / または swap でない場合、行はログに記録されて無視されます。JumpStart <i>fsoptions</i> はサポートされません。</p> <p>サイズの検証は実行されません。このマニフェストを使用して正しくインストールを行うには、生成された AI マニフェスト内で指定されるサイズを調整することが必要な場合があります。</p>
<p>例:</p> <pre>filesys c1t0d0s0 10000 /</pre>	<p>次の AI マニフェストの一部は x86 プラットフォーム用のものです。</p> <pre>&lt;target&gt;   &lt;disk&gt;     &lt;disk_name name="c1t0d0" name_type="ctd"/&gt;     &lt;partition action="create" name="1" part_type="191"&gt;       &lt;slice action="create" force="true"         in_vdev="rpool_vdev" in_zpool="rpool" name="0"&gt;         &lt;size val="10000mb"/&gt;       &lt;/slice&gt;     &lt;/partition&gt;   &lt;/disk&gt;   &lt;logical nodump="true" noswap="false"&gt;     &lt;zpool is_root="true" name="rpool"&gt;       &lt;vdev name="rpool_vdev" redundancy="none"/&gt;     &lt;/zpool&gt;   &lt;/logical&gt; &lt;/target&gt;</pre> <p>js2ai による変換:</p> <p>js2ai コマンドでは、ルートファイルシステム (/) および swap の変換のみサポートします。</p>
<p>例:</p> <pre>filesys mirror:rpool c6t0d0s0 c6t1d0s0 60048 /</pre>	<p>次の AI マニフェストの一部は x86 プラットフォーム用のものです。SPARC プラットフォームの場合、<i>partition</i> 要素は指定されません。</p> <pre>&lt;target&gt;   &lt;disk&gt;     &lt;disk_name name="c6t1d0" name_type="ctd"/&gt;     &lt;partition action="create" name="1" part_type="191"&gt;       &lt;slice action="create" force="true"         in_vdev="rpool_vdev" in_zpool="rpool" name="0"&gt;         &lt;size val="60048mb"/&gt;       &lt;/slice&gt;     &lt;/partition&gt;   &lt;/disk&gt;   &lt;disk&gt;     &lt;disk_name name="c6t0d0" name_type="ctd"/&gt;</pre>

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
	<pre>&lt;partition action="create" name="1" part_type="191"&gt;   &lt;slice action="create" force="true"     in_vdev="rpool_vdev" in_zpool="rpool" name="0"&gt;     &lt;size val="60048mb"/&gt;   &lt;/slice&gt; &lt;/partition&gt; &lt;/disk&gt; &lt;logical nodump="true" noswap="false"&gt;   &lt;zpool is_root="true" name="rpool"&gt;     &lt;vdev name="rpool_vdev" redundancy="mirror"/&gt;   &lt;/zpool&gt; &lt;/logical&gt; &lt;/target&gt;</pre>
<code>filesystem server:path</code>	サポートされていません。AI はリモートファイルシステムをサポートしません。
<code>forced_deployment</code>	サポートされていません。AI はフラッシュアーカイブからインストールしません。
<code>geo</code>	サポートされていません。AI では、言語サポート用の地理的地域はシステム構成プロファイルで指定されます。『 <a href="#">Oracle Solaris 11.2 システムのインストール</a> 』の第 11 章「クライアントシステムの構成」を参照してください。
<code>install_type</code>	AI は <code>initial_install</code> のみサポートします。
<code>layout_constraint</code>	サポートされていません。AI は <code>initial_install</code> のみサポートします。
<code>local_customization</code>	サポートされていません。AI はフラッシュアーカイブからインストールしません。
<code>locale</code>	サポートされていません。AI では、ロケールサポートはシステム構成プロファイルで指定されます。『 <a href="#">Oracle Solaris 11.2 システムのインストール</a> 』の第 11 章「クライアントシステムの構成」を参照してください。
<code>metadb</code>	サポートされていません。
<code>no_content_check</code>	サポートされていません。AI はフラッシュアーカイブからインストールしません。
<code>no_master_check</code>	サポートされていません。AI はフラッシュアーカイブからインストールしません。
<code>num_clients</code>	サポートされていません。
<code>package package</code>	例:
例:	<pre>&lt;software type="IPS"&gt;   &lt;software_data action="install"&gt;     &lt;name&gt;pkg:/SUNWpampkcs11&lt;/name&gt;</pre>
<code>package SUNWpampkcs11 add</code>	

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
	<pre>&lt;/software_data&gt; &lt;/software&gt;</pre> <p>パッケージ名が有効であることを確認する方法については、<a href="#">25 ページの「ソフトウェアパッケージの変換」</a>を参照してください。</p>
<code>package package add nfs server:path</code>	<p>AI マニフェストでは、すべてのソフトウェアは IPS (Image Packaging System) パッケージリポジトリから取得されます。IPS リポジトリの場所は HTTP またはファイルとすることができます。ソフトウェアソースとして複数の IPS リポジトリを指定できます。パブリッシャーの指定方法については、<a href="#">ai_manifest(4)</a> のマニュアルページを参照してください。</p> <p>js2ai による変換:</p> <p>js2ai コマンドは <code>retrieval-type location</code> 引数を無視します。</p>
<code>package package delete</code>	<p>例:</p> <pre>&lt;software type="IPS"&gt;   &lt;software_data action="uninstall"&gt;     &lt;name&gt;pkg:/SUNWpampkcs11&lt;/name&gt;   &lt;/software_data&gt; &lt;/software&gt;</pre> <p>AI は初期インストールのみサポートするため、パッケージを削除するために AI マニフェストを使用する必要はありません。</p>
<code>partitioning default</code> <code>partitioning explicit</code>	<p>ターゲットのインストールデバイスを指定しない場合、AI はデフォルトのインストールデバイスとデフォルトのパーティション分割を決定します。カスタムのパーティション分割を指定するには、<a href="#">ai_manifest(4)</a> のマニュアルページを参照してください。</p> <p>js2ai による変換:</p> <p>js2ai では、パーティション分割は <code>default</code> または <code>explicit</code> とする必要があります。<code>explicit</code> の場合、<code>swap</code> および <code>/</code> のみがサポートされます</p>
<code>partitioning existing</code>	<p>サポートされていません。js2ai では、パーティション分割は <code>default</code> または <code>explicit</code> とする必要があります。</p>
<code>patch</code>	<p>サポートされていません。AI は <code>initial_install</code> のみサポートします。</p> <p>システムまたは特定のパッケージを更新するには、インストールされたシステム上で <code>pkg update</code> コマンドを使用します。</p>
<code>pool newpool auto auto auto device</code>	<p>例:</p> <pre>&lt;target&gt;</pre>

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
	<pre data-bbox="820 384 1383 779"> &lt;disk&gt;   &lt;disk_name name="clt0d0" name_type="ctd"/&gt;   &lt;partition action="create" name="1" part_type="191"&gt;     &lt;slice action="create" force="true"       in_vdev="rpool_vdev" in_zpool="newpool" name="0"/&gt;   &lt;/partition&gt; &lt;/disk&gt; &lt;logical nodump="false" noswap="false"&gt;   &lt;zpool is_root="true" name="newpool"&gt;     &lt;vdev name="rpool_vdev" redundancy="none"/&gt;   &lt;/zpool&gt; &lt;/logical&gt; &lt;/target&gt; </pre> <p data-bbox="820 808 982 835">js2ai による変換:</p> <p data-bbox="820 863 1383 989">プロファイル内でプールが指定された場合、js2ai は指定されたデバイスを使用して ZFS ルートプールを作成します。ZFS ルートプールに使用するデバイスを js2ai が決定する際、pool キーワードが他のすべてのキーワードに優先されます。</p> <p data-bbox="820 1016 1383 1142">js2ai コマンドは、プールサイズ、スワップサイズ、またはダンプサイズの検証を一切実行しません。このマニフェストを使用したインストールに成功するには、生成された AI マニフェスト内のサイズを調整することが必要な場合があります。</p>
<p data-bbox="386 1157 734 1184">pool <i>newpool</i> auto auto auto any</p>	<p data-bbox="820 1157 1383 1318">js2ai で、物理デバイス名の代わりに any を指定する場合、any パラメータを含む指定よりも前にデバイス情報を提供しておく必要があります。たとえば、この pool 指定の前に root_device または usedisk 指定を行うことができます。例については、<a href="#">例 2-6「mirrorpool.profile のエラーの修正」</a>を参照してください。</p>
<p data-bbox="386 1333 597 1360">root_device clt0d0s0</p>	<p data-bbox="820 1333 852 1360">例:</p> <pre data-bbox="820 1388 1383 1770"> &lt;target&gt;   &lt;disk&gt;     &lt;disk_name name="clt0d0" name_type="ctd"/&gt;     &lt;partition action="create" name="1" part_type="191"&gt;       &lt;slice action="create" force="true"         in_vdev="rpool_vdev" in_zpool="rpool" name="0"/&gt;     &lt;/partition&gt;   &lt;/disk&gt;   &lt;logical nodump="true" noswap="false"&gt;     &lt;zpool is_root="true" name="rpool"&gt;       &lt;vdev name="rpool_vdev" redundancy="none"/&gt;     &lt;/zpool&gt;   &lt;/logical&gt; </pre>

JumpStart プロファイルファイルのキーワード	AI マニフェストの指令
	</target>  js2ai による変換:  root_device が指定された場合、js2ai は指定されたデバイスに rootdisk を設定します。
system_type	AI マニフェストはシステムタイプを区別しません。  js2ai による変換:  値 standalone のみがサポートされています。
usedisk	js2ai による変換:  js2ai コマンドは後続の any または rootdisk 指定を解決するために、指定された 1 つ以上のデバイスを使用する場合があります。指定されたがこの目的で使用されないデバイスは、ZFS ルートプールがミラー化されていない場合、js2ai によってこのプールに追加されます。

## js2ai の使用による JumpStart ルールおよびプロファイルから AI 条件およびマニフェストへの変換

js2ai コマンドで `-r` オプションを使用すると、JumpStart ルールおよび関連付けられたプロファイルが AI 条件およびマニフェストに変換されます。最初は `-s` オプションを使用して検証をスキップします。このコマンドは、rules ファイルおよび rules ファイルによって参照されるプロファイルで変換操作を実行します。rules ファイルで参照される各プロファイルは、AI クライアントプロビジョニングマニフェスト (`/usr/share/auto_install/manifest/default.xml`) に対して処理されます。この段階では、JumpStart rules ファイルで指定されたプロファイルごとに、`AI_profile-name` という名前のディレクトリが作成されます。`AI_profile_name` ディレクトリには、このプロファイルを参照するルールに対応する AI 条件ファイルが `criteria-rule-number.xml` の形式で格納されます。また、`AI_profile-name` ディレクトリには、`profile-name` プロファイルファイルに対応する AI マニフェストファイルが `profile-name.arch.xml` の形式で格納されます。

変換が正しく完了したことを示すメッセージを受け取ったら、`-s` オプションを付けずに `js2ai` コマンドを実行して、出力された AI マニフェストを検証します。検証エラーは AI マニフェストファイル内で修正する必要があります。

変換が正常に完了したというメッセージが表示されない場合、エラーレポートおよび `js2ai.log` ファイルを調べます。エラーレポートおよびログファイルは、警告、処理エラー、サポートされない項目、変換エラー、および検証エラーを報告します。エラーレポートは `stdout` への表出力で、`rules` およびプロファイルファイルの変換中に発生したタイプごとのエラーの番号を示します。ログファイルには問題が記述されます。

1. 処理エラーを修正します。
2. `rules` ファイルおよびプロファイルファイルから、サポート外項目として一覧表示されたすべての行を削除します。
3. 変換エラーを検査して、可能な場合はエラーを修正します。それ以外の場合は、エラーの原因となっている行を削除します。
4. 警告メッセージを検査して、修正が必要ないことを確認します。

`js2ai` コマンドの次のオプションを確認してください。

- ルールファイルまたはプロファイルファイルの変換に関する詳細な情報を表示するには、`-v` オプションを使用します。詳細は、[37 ページの「追加のプロファイル変換情報の表示」](#)を参照してください。
- 個々の出力 AI マニフェストを適切な AI DTD に対して検証するには、`-v` オプションを使用します。詳細は、[38 ページの「出力 AI マニフェストの検証」](#)を参照してください。
- 1 つの JumpStart プロファイルファイルのみを変換するには、`-p` オプションを使用します。[32 ページの「AI マニフェストファイルの修正」](#)の例を参照してください。

## js2ai による JumpStart デバイス指定の変換方法

このセクションでは、`js2ai` が JumpStart プロファイル指定から AI マニフェスト用のターゲット要素を決定する方法について説明します。

### システムのルートディスクを決定する方法

プロファイルの変換プロセス中、`js2ai` はプロファイルが参照するクライアントシステムにアクセスしないため、`js2ai` はできるかぎり JumpStart と一致するプロセスを使用して、変換中にルートディスクを決定しようとします。

`js2ai` コマンドは次の段階を実行して、ルートディスクに使用するデバイスを決定します。

1. プロファイルで `root_device` キーワードが指定されている場合、`js2ai` は `rootdisk` をスライズが存在するデバイスに設定します。

2. `rootdisk` が設定されておらず、プロファイルで `boot_device` キーワードが指定されている場合、`js2ai` は `rootdisk` をブートデバイスに設定します。
3. `rootdisk` が設定されておらず、`partitioning default` が指定され、`solaris fdisk` エントリが指定されている場合、`js2ai` は `rootdisk` を指定されたディスク名に設定します。
4. `rootdisk` が設定されておらず、プロファイルで `fileys cwtxdysz size /` エントリが指定されている場合、`js2ai` は `rootdisk` をそのエントリで指定された `cwtxdysz` ディスクに設定します。
5. `rootdisk` が設定されておらず、プロファイルで `usedisk disk-name` エントリが指定されている場合、`js2ai` は `rootdisk` をそのエントリで指定された `disk-name` ディスクに設定します。
6. `rootdisk` が設定されておらず、プロファイルで次の指定が見つかった場合 (`size` は 0 または `delete` でなく、`disk` は `all` でない)、`rootdisk` はこのディスク名に設定されます。

```
fdisk disk-name solaris size
```

7. `rootdisk` が設定されていない場合、デバイスが `rootdisk` として指定されたオカレンスで変換エラーが生成されます。

## any デバイスの変換方法

`js2ai` コマンドは次の段階を実行して、`any` キーワードが指定されているときに使用するデバイスを決定します。

1. `any` デバイスが指定されていて、キーワードアクション (ミラー化されていないプール、または / マウントポイントを使用した `fileys`) が指定されている場合、`any` デバイスは `rootdisk` に設定されます (`rootdisk` が設定されている場合)。
2. `any` デバイスが変換されておらず、`usedisk` 文がプロファイル内に存在する場合、`any` デバイスは `usedisk` 文で指定されたデバイスに設定されます。
3. `any` デバイスが変換されておらず、`any` デバイスが指定されたアクションによって ZFS ルートプールが作成される場合、AI はそのデバイスを選択します (ただし、ミラープールが指定されている場合を除きます)。

## ZFS ルートプールを決定する方法

`js2ai` コマンドは次のステップを実行して、ZFS ルートプールで使用するデバイスを決定します。ZFS ルートプールが決定されると、その後に検出される定義がすでに決定されている ZFS ルートプールと競合する場合、エラーとしてフラグが設定されます。

1. プロファイルで `pool` キーワードが指定されている場合、js2ai は ZFS ルートプールを `pool` キーワードで指定されたデバイスに設定します。
2. ZFS ルートプールが決定されておらず、プロファイルでマウントポイント `/` を使用してファイルシステムが指定された場合、ZFS ルートプールは指定されたデバイスを使用して作成されます。
3. ZFS ルートプールが決定されておらず、プロファイル内のすべてのキーワードが処理され、`rootdisk` が設定されている場合、ZFS ルートプールは `rootdisk` デバイスを使用して作成されます。
4. ZFS ルートプールが決定されておらず、パーティションタイプが `default` の場合、AI は ZFS ルートプールで使用するデバイスを選択します。
5. ZFS ルートプールが決定されておらず、処理中にエラーが発生しなかった場合、AI は ZFS ルートプールで使用するデバイスを選択します。
6. ZFS ルートプールがミラー化されたプールではなく、指定された 1 つ以上の `usedisk` デバイスが `rootdisk` デバイスまたは `any` デバイスの変換で使用されていない場合、これらのディスクが ZFS ルートプールに追加されます。

## ソフトウェアパッケージの変換

JumpStart プロファイルでは `cluster` および `package` キーワードを使用してソフトウェアをシステムにインストールします。`cluster` キーワードは AI ではサポートされません。IPS では、`incorporation` パッケージまたはグループパッケージをインストールするための構文は、その他のパッケージをインストールするための構文と同じなので、JumpStart プロファイルの `cluster` を `package` に変更するだけで、js2ai コマンドは、AI マニフェスト内に正しいパッケージインストール指定を作成します。

---

**ヒント** - AI マニフェスト内のパッケージ名を確認してください。AI マニフェスト内でインストール用に指定されたパッケージが、その AI マニフェスト内で指定されたどのパブリッシャー起点からも利用できない場合、クライアントインストールは失敗します。

---

IPS パッケージ名は SVR4 パッケージ名と異なります。たとえば、SVR4 パッケージ `SUNWpampkcs11` の名前は IPS では `library/security/pam/module/pam-pkcs11` に変更されます。

SVR4 パッケージ名が IPS 内に存在する場合は、SVR4 名を使用して IPS パッケージをインストールできます。たとえば、AI マニフェストで `SUNWpampkcs11` パッケージのインストールが指

定された場合、`library/security/pam/module/pam-pkcs11` パッケージが自動的にインストールされます。この場合、パッケージの名前が変更されます。

SVR4 パッケージ名が IPS 内に存在しない場合は、パッケージ名を変更するか、その指定を AI マニフェストから削除する必要があります。たとえば、`SUNWCall` および `SUNWCuser` パッケージは IPS 内で名前変更されません。AI マニフェストでこれらのパッケージが指定された場合、インストールは失敗します。

AI マニフェスト内で特定のパッケージ名が使用できるかどうかを判断するには、Oracle Solaris 11 システムで `pkg list` コマンドを使用します。AI マニフェスト内で指定された IPS パッケージリポジトリ起点からのパッケージを一覧表示するための `-g` オプションを必ず使用してください。

`js2ai` コマンドは、JumpStart プロファイルファイルからの指定を含む新規 AI マニフェストを構築するための基礎として、`/usr/share/auto_install/manifest/default.xml` AI マニフェストを使用します。このデフォルト AI マニフェストは、基本オペレーティングシステムをインストールする 2 つのパッケージ (`entire` および `solaris-large-server`) のインストールを指定します。これら 2 つのパッケージのほかは、おそらく追加のツールおよびアプリケーションのインストールを指定するだけで済みます。

#### 例 2-1 パッケージ名が使用できるかどうかの確認

この例では、AI マニフェストは `http://pkg.oracle.com/solaris/release` リポジトリ起点を指定します。

```
$ pkg list -af -g http://pkg.oracle.com/solaris/release SUNWCall SUNWCuser
pkg list: no packages matching 'SUNWCuser, SUNWCall' known
```

このメッセージは、これら 2 つのパッケージがこの AI マニフェスト内で使用できないことを示しています。

#### 例 2-2 名前が変更されたパッケージの操作

`pkg list` コマンドを使用して、名前が変更されたパッケージを特定することもできます。

```
$ pkg list -af -g http://pkg.oracle.com/solaris/release SUNWpampkcs11
NAME (PUBLISHER)                VERSION                IFO
SUNWpampkcs11                   0.6.0-0.133          --r
```

最後の列の「r」は、このパッケージの名前が変更されていることを示します。この名前を AI マニフェスト内で使用できますが、`pkg info` コマンドを使用してパッケージの新しい名前を決定した方がよい場合もあります。

次の出力の「Renamed to」行を参照してください。SUNWpampkcs11 パッケージの名前は library/security/pam/module/pam-pkcs11 に変更されています。今後の Oracle Solaris 更新との互換性を高めるために、AI マニフェスト内で library/security/pam/module/pam-pkcs11 を指定した方がよい場合もあります。

```
$ pkg info -r SUNWpampkcs11
    Name: SUNWpampkcs11
    Summary:
    State: Not installed (Renamed)
    Renamed to: library/security/pam/module/pam-pkcs11@0.6.0-0.133
               consolidation/sfw/sfw-incorporation
    Publisher: solaris
    Version: 0.6.0
    Build Release: 5.11
    Branch: 0.133
    Packaging Date: Wed Oct 27 18:50:11 2010
    Size: 0.00 B
    FMRI: pkg://solaris/SUNWpampkcs11@0.6.0,5.11-0.133:20101027T185011Z

$ pkg info -r pam-pkcs11
    Name: library/security/pam/module/pam-pkcs11
    Summary: The OpenSC PKCS#11 PAM Login Tools
    Category: System/Security
    State: Not installed
    Publisher: solaris
    Version: 0.6.0
    Build Release: 5.11
    Branch: 0.175.2.0.0.27.0
    Packaging Date: Mon Nov 11 17:47:35 2013
    Size: 1.74 MB
    FMRI: pkg://solaris/library/security/pam/module/pam-pkcs11@0...
```

### 例 2-3 廃止されたパッケージの操作

AI マニフェスト内で SUNWmysql を database/mysql-51 に置換した方がよい場合もあります。

```
$ pkg list -af -g http://pkg.oracle.com/solaris/release SUNWmysql
NAME (PUBLISHER)                VERSION                IFO
SUNWmysql                       4.0.24-0.142         --o
```

最後の列の「o」は、このパッケージが廃止されていることを示します。このパッケージ名は AI マニフェスト内で使用できません。pkg list コマンドにワイルドカードを指定して使用するか、pkg search コマンドを使用して、使用可能な別の名前でもパッケージが使用できるかどうかを判断します。

```
$ pkg list -af SUNWmysql*
NAME (PUBLISHER)                VERSION                IFO
SUNWmysql                       4.0.24-0.142         --o
SUNWmysql-base                 0.5.11-0.133         --r
SUNWmysql-python               0.5.11-0.162         --o
```

```

SUNWmysql-python26                0.5.11-0.133      --r
SUNWmysql5                          5.0.86-0.171      --o
SUNWmysql5                          5.0.86-0.133      --r
SUNWmysql51                         5.1.37-0.133      --r
SUNWmysql51lib                      5.1.37-0.133      --r
SUNWmysql51test                    5.1.37-0.133      --r
SUNWmysql5jdbc                     5.1.5-0.171       --o
SUNWmysql5jdbc                     5.1.5-0.133       --r
SUNWmysql5test                     5.0.86-0.171      --o
SUNWmysql5test                     5.0.86-0.133      --r
SUNWmysqlt                          4.0.24-0.142      --o
$ pkg info -r SUNWmysql51
Name: SUNWmysql51
Summary:
State: Not installed (Renamed)
Renamed to: database/mysql-51@5.1.37-0.133
            consolidation/sfw/sfw-incorporation
Publisher: solaris
Version: 5.1.37
Build Release: 5.11
Branch: 0.133
Packaging Date: Wed Oct 27 18:49:18 2010
Size: 0.00 B
FMRI: pkg://solaris/SUNWmysql51@5.1.37,5.11-0.133:20101027T184918Z

```

AI マニフェスト内で SUNWmysql を database/mysql-51 に置換した方がよい場合もあります。

## js2ai を使用したルールおよびプロファイルの変換

このセクションでは、単一の js2ai コマンドの使用による、JumpStart rules ファイルおよびその rules ファイルによって参照されるすべてのプロファイルファイルの変換を示します。デフォルトでは、変換された各プロファイルは、AI\_profile-filename という名前のディレクトリに出力されます。そのプロファイルを選択するルールは、AI 基準ファイルと同じディレクトリに出力されません。

### JumpStart ルールファイルとプロファイルファイルの例

この例では次の rules ファイルを使用します。

```

# The following rule matches only one system:

hostname sample_host    - fdisk.profile          -

# The following rule matches only one system:

hostaddress 10.6.68.127 - mirrorfilesys.profile -

```

```
# The following rule matches any system that is on the 924.222.43.0 network:
```

```
network 924.222.43.0 - rootdisk.profile -
```

```
# The following rule matches all x86 systems:
```

```
arch i386 - mirrorpool.profile -
```

fdisk.profile ファイルの内容は次のとおりです。

```
install_type initial_install
system_type server
root_device clt0d0s0
usedisk clt0d0
fdisk rootdisk solaris all
partitioning explicit
fileys rootdisk.s1 5000 swap
fileys rootdisk.s0 10000 /
cluster SUNWCall
```

mirrorfileys.profile ファイルの内容は次のとおりです。

```
install_type initial_install
partitioning default
fileys mirror c6t0d0s0 c6t1d0s0 60048 /
cluster SUNWCuser
```

mirrorpool.profile ファイルの内容は次のとおりです。

```
install_type initial_install
partitioning default
pool newpool auto auto auto mirror any any
cluster SUNWCuser
```

rootdisk.profile ファイルの内容は次のとおりです。

```
install_type initial_install
partitioning explicit
fileys rootdisk.s0 15000 /
fileys rootdisk.s1 1000 swap
cluster SUNWCall
```

## js2ai を使用して rules ファイルと関連プロファイルを処理する

次のコマンドを使用して、この rules ファイルを処理します。検証が実行されなかったため、エラーレポートでは検証エラーはハイフン文字として表示されます。検証は -s オプションによって抑制されます。

```
# js2ai -rS
Name                               Process Warnings Errors  Unsupported Items  Conversion Errors  Validation Errors
```

```

-----
rules                0      0      1      0      -
fdisk.profile        0      0      2      0      -
mirrorfileys.profile 0      0      2      0      -
mirrorpool.profile   0      0      1      1      -
rootdisk.profile     0      0      1      2      -

Conversion completed. One or more failures and/or warnings occurred.
For errors see js2ai.log
# cat js2ai.log
rules:line 3:UNSUPPORTED: unsupported keyword: hostname
fdisk.profile:line 2:UNSUPPORTED: unsupported value for 'system_type' specified: server
fdisk.profile:line 9:UNSUPPORTED: unsupported keyword: cluster
mirrorfileys.profile:line 3:UNSUPPORTED: unsupported mount point of 'unnamed' specified,
mount points other than '/' and 'swap' are not supported
mirrorfileys.profile:line 4:UNSUPPORTED: unsupported keyword: cluster
mirrorpool.profile:line 3:CONVERSION: unable to convert 'any' device to physical device.
  Replace 'any'
with actual device name
mirrorpool.profile:line 4:UNSUPPORTED: unsupported keyword: cluster
rootdisk.profile:line 3:CONVERSION: unable to convert 'rootdisk.s0'.  Replace 'rootdisk.' with
actual
device name
rootdisk.profile:line 4:CONVERSION: unable to convert 'rootdisk.s1'.  Replace 'rootdisk.' with
actual
device name
rootdisk.profile:line 5:UNSUPPORTED: unsupported keyword: cluster

```

出力は、*AI\_profile-filename* という名前のディレクトリに格納されます。JumpStart ルールから作成されたすべての AI 条件ファイルは、rules ファイル内のルールから名前が付けられます。AI マニフェストには *profile-filename.arch.xml* という名前が付けられ、*arch* は generic、x86、または sparc です。

```

# ls AI_*
AI_fdisk.profile:
fdisk.profile.x86.xml

AI_mirrorfileys.profile:
criteria-2.xml      mirrorfileys.profile.generic.xml

AI_mirrorpool.profile:
criteria-2.xml      mirrorpool.profile.generic.xml

AI_rootdisk.profile:
criteria-3.xml      rootdisk.profile.generic.xml

```

## hostname キーワードの置換

ログファイルでは、JumpStart ルールキーワード `hostname` が、AI マニフェストを選択するための AI 条件キーワードとしてサポートされないと報告されています。`hostname` キーワード

は、`fdisk.profile` プロファイルを使用するクライアントを指定します。`hostname` は AI マニフェストを選択するためにサポートされるキーワードではないため、新しい `AI_fdisk.profile` ディレクトリには AI 条件ファイルが含まれません。

この問題は、`rules` ファイル内で `hostname` を `hostaddress` に変更することによって修正できます。

この問題は、`hostname` システムを識別する MAC アドレスまたは IP アドレスを指定する AI 条件ファイルを作成することによっても修正できます。たとえば、`sample_host` の MAC アドレスが `0:14:4F:20:53:97` の場合、次の条件ファイルは JumpStart ルール `hostname sample_host` と同等です。

```
<?xml version="1.0" encoding="utf-8"?>
<ai_criteria_manifest>
  <ai_criteria name="mac">
    <value>0:14:4F:20:53:97</value>
  </ai_criteria>
</ai_criteria_manifest>
```

システムの MAC アドレスを確認するには、[dladm\(1M\)](#) のマニュアルページに説明されている `dladm` コマンドを使用します。

JumpStart ルール `hostaddress 10.6.68.127` について、`js2ai` コマンドは AI 条件ファイル `AI_mirrorfilesys.profile/criteria-2.xml` を自動的に作成し、JumpStart の `hostaddress` キーワードを AI の `ipv4` キーワードに置換します。

```
<?xml version="1.0" encoding="utf-8"?>
<ai_criteria_manifest>
  <ai_criteria name="ipv4">
    <value>
      10.6.68.127
    </value>
  </ai_criteria>
</ai_criteria_manifest>
```

JumpStart ルール `network 924.222.43.0` について、`js2ai` コマンドは AI 条件ファイル `AI_rootdisk.profile/criteria-3.xml` を自動的に作成し、指定されたネットワークアドレスに基づいて IP アドレスの範囲を指定します。

```
<?xml version="1.0" encoding="utf-8"?>
<ai_criteria_manifest>
  <ai_criteria name="ipv4">
    <range>
      924.222.43.0 924.222.43.255
    </range>
  </ai_criteria>
</ai_criteria_manifest>
```

```
</ai_criteria>
</ai_criteria_manifest>
```

JumpStart ルール arch i386 について、js2ai コマンドは AI 条件ファイル AI\_mirrorpool.profile/criteria-4.xml を自動的に作成し、JumpStart の arch キーワードを AI の cpu キーワードに置換します。

```
<?xml version="1.0" encoding="utf-8"?>
<ai_criteria_manifest>
  <ai_criteria name="cpu">
    <value>
      i386
    </value>
  </ai_criteria>
</ai_criteria_manifest>
```

## AI マニフェストファイルの修正

js2ai コマンドは通常、エラーが報告される場合であっても、JumpStart プロファイルごとに AI マニフェストを作成します。このセクションでは、出力される AI マニフェストがさらに完全になるようにするために、一部の一般的なエラーに対処する方法について説明します。

### 例 2-4 fdisk.profile のエラーの修正

js2ai コマンドが fdisk.profile JumpStart プロファイルについて次のエラーを表示しました。

```
fdisk.profile:line 2:UNSUPPORTED: unsupported value for 'system_type' specified: server
fdisk.profile:line 9:UNSUPPORTED: unsupported keyword: cluster
```

これら 2 つの行は無視され、出力 AI マニフェストに影響しません。変換でエラーメッセージを出さないようにする場合は、これら 2 つの行を削除できます。fdisk.profile ファイルは次のような内容になります。

```
install_type initial_install
root_device clt0d0s0
usedisk clt0d0
fdisk rootdisk solaris all
partitioning explicit
filesys rootdisk.s1 5000 swap
filesys rootdisk.s0 10000 /
```

js2ai コマンドの -p オプションを使用して、このプロファイルのみ処理します。

```
# js2ai -p fdisk.profile
```

Successfully completed conversion

出力 AI マニフェスト AI\_fdisk.profile/fdisk.profile.x86.xml の target スタンザには次の内容があります。

```
<target>
  <disk whole_disk="true">
    <disk_name name="c1t0d0" name_type="ctd"/>
    <partition action="create" name="1" part_type="191">
      <slice name="1" action="create" force="true" is_swap="true">
        <size val="5000mb"/>
      </slice>
      <slice name="0" action="create" force="true" in_zpool="rpool" in_vdev="rpool_vdev">
        <size val="10000mb"/>
      </slice>
    </partition>
  </disk>
  <logical noswap="false" nodump="true" >
    <zpool name="rpool" is_root="true">
      <vdev name="rpool_vdev" redundancy="none"/>
      <filesystem name="export" mountpoint="/export"/>
      <filesystem name="export/home"/>
      <be name="solaris"/>
    </zpool>
  </logical>
</target>
```

software スタンザは /usr/share/auto\_install/manifest/default.xml 内のものと同じです。これは、このプロファイルにパッケージ指定が含まれていないためです。

#### 例 2-5 mirrorfilesystems.profile のエラーの修正

js2ai コマンドが mirrorfilesystems.profile JumpStart プロファイルについて次のエラーを表示しました。

```
mirrorfilesystems.profile:line 3:UNSUPPORTED: unsupported mount point of 'unnamed' specified,
mount points other than '/' and 'swap' are not supported
mirrorfilesystems.profile:line 4:UNSUPPORTED: unsupported keyword: cluster
```

JumpStart プロファイルファイルでは、filesystems mirror 指定の *file\_system* パラメータは省略可能です。js2ai 変換では、*file\_system* パラメータは省略できず、/ または swap の 2 つの値のいずれかを持つ必要があります。

mirrorfilesystems.profile ファイルを編集して、filesystems 指定の末尾に「/」を追加し、cluster 行を削除します。mirrorfilesystems.profile ファイルは次のような内容になります。

```
install_type initial_install
partitioning default
filesystems mirror c6t0d0s0 c6t1d0s0 60048 /
```

AI\_mirrorfilesys.profile/criteria-2.xml ファイルを別の場所に保存します。次に、js2ai コマンドの -p オプションを使用して、このプロファイルのみ処理します。

```
# js2ai -p mirrorfilesys.profile
Successfully completed conversion
```

AI\_mirrorfilesys.profile ディレクトリには、mirrorfilesys.profile.sparc.xml および mirrorfilesys.profile.x86.xml の 2 つの出力 AI マニフェストが格納されています。これら 2 つの AI マニフェストの違いは、mirrorfilesys.profile.x86.xml には 2 つの partition 指定が含まれていることです。mirrorfilesys.profile.x86.xml マニフェストの target スタンザの内容は次のとおりです。

```
<target>
  <disk>
    <disk_name name="c6t1d0" name_type="ctd"/>
    <partition action="create" name="1" part_type="191">
      <slice action="create" force="true" in_vdev="rpool_vdev" in_zpool="rpool" name="0">
        <size val="60048mb"/>
      </slice>
    </partition>
  </disk>
  <disk>
    <disk_name name="c6t0d0" name_type="ctd"/>
    <partition action="create" name="1" part_type="191">
      <slice action="create" force="true" in_vdev="rpool_vdev" in_zpool="rpool" name="0">
        <size val="60048mb"/>
      </slice>
    </partition>
  </disk>
  <logical noswap="false" nodump="false" >
    <zpool name="newpool" is_root="true">
      <vdev name="rpool_vdev" redundancy="mirror"/>
      <filesystem name="export" mountpoint="/export"/>
      <filesystem name="export/home">
        <be name="solaris">
      </zpool>
    </logical>
  </target>
```

software スタンザは /usr/share/auto\_install/manifest/default.xml 内のものと同じです。これは、このプロファイルにパッケージ指定が含まれていないためです。

#### 例 2-6 mirrorpool.profile のエラーの修正

js2ai コマンドが mirrorpool.profile JumpStart プロファイルについて次のエラーを表示しました。

```
mirrorpool.profile:line 4:CONVERSION: unable to convert 'any' device to physical device.
```

```
Replace 'any' with actual device name
mirrorpool.profile:line 8:UNSUPPORTED: unsupported keyword: cluster
```

これらのエラーを修正するには、次の 2 つの方法のいずれかを使用します。これらの修正は、両方ともまったく同じ AI マニフェスト出力を生成します。[24 ページの「any デバイスの変換方法」](#)も参照してください。

- `mirrorpool.profile` プロファイルを編集し、`any` エントリを物理デバイス名で置換します。また、`cluster` 行を削除します。`mirrorpool.profile` ファイルの内容は次のようになります。

```
install_type initial_install
partitioning default
pool newpool auto auto auto mirror c6t0d0s0 c6t1d0s0
```

- `pool` 指定の `vdev-list` 一覧に、物理デバイス名でなく `any` を指定した場合、`any` パラメータを含む指定よりも前にデバイス情報を提供しておく必要があります。`mirrorpool.profile` プロファイルを編集して、プール指定の前に `usedisk` 指定を追加します。また、`cluster` 行を削除します。`mirrorpool.profile` ファイルの内容は次のようになります。

```
install_type initial_install
partitioning default
usedisk c6t0d0 c6t1d0
pool newpool auto auto auto mirror any any
```

`AI_mirrorpool.profile/criteria-4.xml` ファイルを別の場所に保存します。次に、`js2ai` コマンドの `-p` オプションを使用して、このプロファイルのみ処理します。

```
# js2ai -p mirrorpool.profile
Successfully completed conversion
```

`AI_mirrorpool.profile` ディレクトリには、`mirrorpool.profile.sparc.xml` および `mirrorpool.profile.x86.xml` の 2 つの出力 AI マニフェストが格納されています。これら 2 つの AI マニフェストの違いは、`mirrorpool.profile.x86.xml` には 2 つの `partition` 指定が含まれていることだけです。`mirrorpool.profile.x86.xml` マニフェストの `target` スタンザの内容は次のとおりです。

```
<target>
  <disk>
    <disk_name name="c6t1d0" name_type="ctd"/>
    <partition action="create" name="1" part_type="191">
      <slice action="create" force="true" in_vdev="rpool_vdev" in_zpool="newpool" name="0"/>
    </partition>
  </disk>
</disk>
```

```

<disk_name name="c0t0d0" name_type="ctd"/>
<partition action="create" name="1" part_type="191">
  <slice action="create" force="true" in_vdev="rpool_vdev" in_zpool="newpool" name="0"/>
</partition>
</disk>
<logical noswap="false" nodump="true">
  <zpool name="rpool" is_root="true">
    <vdev name="rpool_vdev" redundancy="mirror"/>
    <filesystem name="export" mountpoint="/export"/>
    <filesystem name="export/home">
      <be name="solaris">
    </zpool>
  </logical>
</target>

```

software スタンザは /usr/share/auto\_install/manifest/default.xml 内のものと同じです。これは、このプロファイルにパッケージ指定が含まれていないためです。

#### 例 2-7 rootdisk.profile のエラーの修正

js2ai コマンドが rootdisk.profile JumpStart プロファイルについて次のエラーを表示しました。

```

rootdisk.profile:line 3:CONVERSION: unable to convert 'rootdisk.s0'.
Replace 'rootdisk.' with actual device name
rootdisk.profile:line 4:CONVERSION: unable to convert 'rootdisk.s1'.
Replace 'rootdisk.' with actual device name
rootdisk.profile:line 5:UNSUPPORTED: unsupported keyword: cluster

```

これらのエラーを修正するには、次の 2 つの方法のいずれかを使用します。これらの修正は、両方ともまったく同じ AI マニフェスト出力を生成します。

- rootdisk.profile プロファイルを編集し、rootdisk. を c0t0d0 で置換します。また、cluster 行を削除します。rootdisk.profile ファイルは次のような内容になります。

```

install_type initial_install
partitioning explicit
fileys c0t0d0s0 15000 /
fileys c0t0d0s1 1000 swap

```

- rootdisk.profile プロファイルを編集して root\_device 指定を追加し、cluster 行を削除します。rootdisk.profile ファイルは次のような内容になります。

```

install_type initial_install
partitioning explicit
root_device c0t0d0s0
fileys rootdisk.s0 15000 /

```

```
fileys rootdisk.s1 1000 swap
```

AI\_rootdisk.profile/criteria-3.xml ファイルを別の場所に保存します。次に、js2ai コマンドの -p オプションを使用して、このプロファイルのみ処理します。

```
# js2ai -p rootdisk.profile
Successfully completed conversion
```

AI\_rootdisk.profile ディレクトリには、rootdisk.profile.sparc.xml および rootdisk.profile.x86.xml の 2 つの出力 AI マニフェストが格納されています。これら 2 つの AI マニフェストの違いは、rootdisk.profile.x86.xml に partition 指定が含まれていることです。rootdisk.profile.x86.xml マニフェストの target スタンザの内容は次のとおりです。

```
<target>
  <disk>
    <disk_name name="c0t0d0" name_type="ctd"/>
    <partition action="create" name="1" part_type="191">
      <slice action="create" force="true" in_vdev="rpool_vdev" in_zpool="rpool" name="0">
        <size val="15000mb"/>
      </slice>
      <slice action="create" force="true" is_swap="true" name="1">
        <size val="1000mb"/>
      </slice>
    </partition>
  </disk>
  <logical noswap="false" nodump="true" noswap="false">
    <zpool name="rpool" is_root="true" >
      <vdev name="rpool_vdev" redundancy="none"/>
      <filesystem name="export" mountpoint="/export"/>
      <filesystem name="export/home">
        <be name="solaris">
      </zpool>
    </logical>
  </target>
```

software スタンザは /usr/share/auto\_install/manifest/default.xml 内のものと同じです。これは、このプロファイルにパッケージ指定が含まれていないためです。

## 追加のプロファイル変換情報の表示

ルールまたはプロファイル変換あるいは AI マニフェスト検証に関する追加情報を表示するには、js2ai コマンドに -v オプションを指定します。-v オプションを指定すると、処理ステップが表示され、エラーレポートには rules ファイルまたはプロファイルをレポートから省略するのではなく、ゼロが表示されます。

```
# js2ai -v -p rootdisk.profile
Processing profile: rootdisk.profile
Performing conversion on: rootdisk.profile
Generating x86 manifest for: rootdisk.profile
Validating rootdisk.profile.x86.xml
Generating sparc manifest for: rootdisk.profile
Validating rootdisk.profile.sparc.xml
```

Name	Warnings	Process Errors	Unsupported Items	Conversion Errors	Validation Errors
-----	-----	-----	-----	-----	-----
rootdisk.profile	0	0	0	0	0

Successfully completed conversion

## 出力 AI マニフェストの検証

js2ai コマンドを実行した結果得られる AI マニフェストを検証するには、-v オプションを使用します。

```
# js2ai -V ./AI_rootdisk.profile/rootdisk.profile.sparc.xml
Successfully completed conversion
# js2ai -v -V ./AI_rootdisk.profile/rootdisk.profile.sparc.xml
Validating rootdisk.profile.sparc.xml
```

Name	Warnings	Process Errors	Unsupported Items	Conversion Errors	Validation Errors
-----	-----	-----	-----	-----	-----
rootdisk.profile.sparc	-	-	-	-	0

Successfully completed conversion

# ◆◆◆ 第 3 章

## sysidcfg 構成ファイルの変換

この章では、sysidcfg ファイルを AI システム構成プロファイルに変換する方法を示します。変換の大部分は、js2ai コマンドを使用して実行できます。js2ai コマンドの詳細については、[js2ai\(1M\)](#)のマニュアルページを参照してください。

AI システム構成プロファイルは、システム構成を指定する SMF XML プロファイルファイルです。AI システム構成プロファイルの詳細については、『[Oracle Solaris 11.2 システムのインストール](#)』の第 11 章「[クライアントシステムの構成](#)」を参照してください。

## sysidcfg ファイルキーワードとシステム構成プロファイル指令の比較

次の表では、sysidcfg ファイルのキーワードと AI システム構成プロファイル指定の例を比較します。異なる場合では、異なる SMF プロパティ値が適用されることもあります。43 ページの「[js2ai を使用した sysidcfg の変換](#)」および『[Oracle Solaris 11.2 システムのインストール](#)』の「[システム構成プロファイルの例](#)」の例を参照してください。

表 3-1 sysidcfg ファイルキーワードと構成プロファイル指令の比較

sysidcfg ファイルのキーワード	システム構成プロファイルの指令
keyboard	js2ai コマンドは変換を実行しません。sysidcfg ファイルで指定されたキーボードが Oracle Solaris 11 でサポートされていることを確認してください。  サービス名: system/keymap プロパティグループ名: keymap プロパティ値名: layout
name_service	AI は DNS、NIS、および LDAP をサポートします。js2ai コマンドは NONE、DNS、NIS、および LDAP の値をサポートします。NIS+ ネームサービスは NIS として変換されます。ネームサービスが構成されている場合は、ネットワークを DefaultFixed として構成する必要があります。  サービス名: system/name-service/switch

sysidcfg ファイルのキーワード	システム構成プロファイルの指令
	プロパティグループ名: config プロパティ値名: netgroup
network_interface	<p>AI では、システムインストールの一部として単一インタフェースの構成のみがサポートされます。この制限のため、js2ai コマンドは PRIMARY という名前のインタフェースまたは sysidcfg ファイル内で見つかった最初のインタフェースのみを処理します。name_service が指定されている場合、js2ai コマンドはネットワークを DefaultFixed に設定します。適切に構成された DefaultFixed ネットワークは、ホスト名、IP アドレス、ネットマスク、およびゲートウェイを提供する必要があります。自動ネットワーク構成は、ネームサービスが指定されていない場合にのみサポートされます。</p> <p>Oracle Solaris では、デフォルトでリンク名の割り当て時に接頭辞 net が使用されます。Oracle Solaris 10 の古いスタイルは使用できますが、デフォルトでは無効になっています。js2ai コマンドは、ネットワークインタフェースの変換を実行して、Solaris 10 スタイルのリンク名を見つけると、ニュートラルリンク名を無効にします。Oracle Solaris 11 でニュートラルリンク名を使用するには、sysidcfg ファイルに指定されたネットワークインタフェース名を net0 などの Oracle Solaris ニュートラルリンク名に変更する必要があります。ニュートラルリンク名についての詳細は、『Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理』の「Oracle Solaris のネットワークデバイスとデータリンク名」を参照してください。</p>
nfs4_domain	サービス名: network/install プロパティグループ名: install_ipv4_interface プロパティ値名: static_address プロパティグループ名: install_ipv6_interface <p>nfs4_domain=dynamic は、Automatic ネットワークおよび DefaultFixed ネットワークについてサポートされています。networks.nfs4_domain=&lt;custom-domain-name&gt; は DefaultFixed ネットワークでのみサポートされています。network_interface キーワードの変換によって、DefaultFixed または Automatic のどちらのネットワークが構成されるかが決まります。</p>
root_password	サービス名: network/nfs/mapid プロパティグループ名: nfs-props プロパティ値名: nfsmapid_domain <p>Oracle Solaris 11 では、root ユーザーではなく役割が使用されます。マルチユーザーモードでシステムにアクセスするには、root 役割特権を持つユーザーを定義する必要があります。必要な構造体を root_password キーワードで完全に定義することができないため、root 役割特権を持つユーザーアカウントを定義するために使用される必要な構造体を sc_profile.xml ファイルにコメントとして追加する必要があります。root_password キーワードを指定しない場合は、それに代わる必要なデータ構造体をコメントとして定義する必要があります。</p>
	サービス名: system/config-user プロパティグループ名: root_account プロパティ値名: password

sysidcfg ファイルのキーワード	システム構成プロファイルの指令
security_policy	js2ai コマンドは値 NONE をサポートします。
service_profile	js2ai ツールは値 limited_net をサポートします。
system_locale	js2ai コマンドは、指定されたロケールがサポートされているデフォルトコアロケールの 1 つであることを確認します。コアロケールの詳細については、『 <a href="#">Oracle Solaris 10 から Oracle Solaris 11.2 への移行</a> 』の「 <a href="#">国際化とローカリゼーションの変更</a> 」を参照してください。指定したロケールがコアロケール内にはない場合は、警告が生成されません。js2ai コマンドは、非コアロケールからコアロケールへの変換も試みます。  サービス名: system/environment プロパティグループ名: environment プロパティ値名: LC_ALL
terminal	js2ai コマンドは変換を実行しません。sysidcfg ファイルで指定された端末タイプが Oracle Solaris 11 でサポートされていることを確認してください。  サービス名: system/console-login プロパティグループ名: ttymon プロパティ値名: terminal_type
timeserver	js2ai コマンドは値 localhost をサポートします。
timezone	js2ai コマンドは変換を実行しません。  サービス名: system/timezone プロパティグループ名: timezone プロパティ値名: localtime

## js2ai を使用した sysidcfg ファイルからシステム構成プロファイルへの変換

js2ai コマンドに `-s` オプションを指定して使用して、この JumpStart 構成に関連付けられた sysidcfg ファイルをシステム構成プロファイルファイルに変換します。最初は `-s` オプションを使用して検証をスキップします。

js2ai は、処理された sysidcfg ファイルごとに、js2ai コマンドが起動されたディレクトリ内に `sc_profile.xml` という名前の AI システム構成プロファイルファイルを作成します。sc\_profile.xml ファイルに別のディレクトリを指定するには、`-D` オプションを使用します。

変換が正しく完了したことを示すメッセージを受け取ったら、`-s` オプションを付けずに js2ai コマンドを実行して、出力された sc\_profile.xml ファイルを検証します。検証エラーは sc\_profile.xml ファイル内で修正する必要があります。

変換が正常に完了したというメッセージが表示されない場合、エラーレポートおよび `js2ai.log` ファイルを調べます。エラーレポートおよびログファイルは、警告、処理エラー、サポートされない項目、変換エラー、および検証エラーを報告します。エラーレポートは `stdout` への表出力で、`sysidcfg` ファイルの変換中に発生したタイプごとのエラーの番号を示します。ログファイルには問題が記述されます。

1. 処理エラーを修正します。
2. `sysidcfg` ファイルから、サポート外項目として一覧表示されたすべての行を削除します。
3. 変換エラーを検査して、可能な場合はエラーを修正します。それ以外の場合は、エラーの原因となっている行を削除します。
4. 警告メッセージを検査して、修正が必要ないことを確認します。

変換について詳細な情報を表示するには、`js2ai` コマンドに `-v` を付けて使用します。詳細は、[45 ページの「詳細な構成変換情報の表示」](#)を参照してください。

特定の出力システム構成プロファイルを検証するには、`-v` オプションを付けて `js2ai` コマンドを実行します。[46 ページの「出力システム構成プロファイルの検証」](#)の例を参照してください。

```
# js2ai -v path/sc_profile.xml
```

## js2ai 変換の警告

`sysidcfg` ファイルを `js2ai` で変換すると常に、次の警告が生成されます。

```
sysidcfg:line #:WARNING: Oracle Solaris 11 uses roles instead of root
user. An admin user with root role privileges will need to be defined in order
to access the system in multi-user mode. The necessary xml structures have
been added to sc_profile.xml as a comment. Edit sc_profile.xml to perform
the necessary modifications to define the admin user.
```

この警告は、`js2ai` コマンドが、`root_passwd` キーワードだけでは必要なデータ構造を生成できないことに関係があります。`sysidcfg` ファイルに `root_passwd` キーワードが含まれている場合、`js2ai` は、次のような `system/config-user` 構造体を生成します。

```
<service name="system/config-user" type="service" version="1">
  <instance enabled="true" name="default">
    <!--
Configures user account as follows:
* User account name 'jack'
* password 'jack'
* GID 10
* UID 101
```

```

* root role
* bash shell
-->
  <!--
<property_group name="user_account" type="application">
  <propval name="login" type="astring" value="jack"/>
  <propval name="password" type="astring" value="9Nd/cwBcNWFZg"/>
  <propval name="description" type="astring" value="default_user"/>
  <propval name="shell" type="astring" value="/usr/bin/bash"/>
  <propval name="gid" type="astring" value="10"/>
  <propval name="uid" type="astring" value="101"/>
  <propval name="type" type="astring" value="normal"/>
  <propval name="roles" type="astring" value="root"/>
  <propval name="profiles" type="astring" value="System Administrator"/>
  <propval name="sudoers" type="astring" value="ALL=(ALL) ALL"/>
</property_group>
-->
  <property_group name="root_account" type="application">
    <propval name="login" type="astring" value="root"/>
    <propval name="password" type="astring" value="{your_root_passwd}"/>
    <propval name="type" type="astring" value="role"/>
  </property_group>
</instance>
</service>

```

XML では、コメントは `<!--` で始まります。Oracle Solaris 11 をサポートするために必要なユーザーを定義するには、`user_account` プロパティグループ構造体を囲む XML コメントマーカーを削除します。次に、`user_account` プロパティグループ構造体を変更し、`root` 役割特権を持つユーザーを作成します。`root_password` キーワードを指定しなかった場合は、`root_account` プロパティグループ構造体をコメントにします。

## js2ai を使用した sysidcfg の変換

このセクションでは、`js2ai` を使用して `sysidcfg` ファイルを AI システム構成プロファイルファイルに変換する方法を示します。`js2ai` は、処理された `sysidcfg` ファイルごとに、`js2ai` コマンドが起動されたディレクトリ内に `sc_profile.xml` という名前の AI システム構成プロファイルファイルを作成します。`sc_profile.xml` ファイルに別のディレクトリを指定するには、`-D` オプションを使用します。変換プロセスに役に立つように、元の `sysidcfg` ファイルのコピーを作成し、コピーを使用します。変換プロセスでは、`sysidcfg` ファイルを編集および変更しないと、ファイルを正しく変換できない場合があります。

### 例 3-1 エラーを含む sysidcfg ファイルの変換

この例では、次の `sysidcfg` ファイルを使用します。

```

timezone=US/Pacific
timeserver=localhost
keyboard=US-English
system_locale=en_US.UTF-8
terminal=vt100
network_interface=primary { hostname=host1
    ip_address=192.0.2.4
    netmask=255.255.255.224
    protocol_ipv6=yes
    default_route=192.0.2.3 }
name_service=DNS{domain_name=example.com
    name_server=192.0.2.1,192.0.2.2
    search=example.com,example.org}
root_password=rJmv5LUXM10cU
security_policy=none
nfs_domain=dynamic

```

次のコマンドを使用して、この sysidcfg ファイルを処理します。-s オプションによって検証を抑制したため、エラーレポートでは、検証エラーはハイフン文字として表示されます。

```
# js2ai -sS
```

Name	Warnings	Process Errors	Unsupported Items	Conversion Errors	Validation Errors
sysidcfg	1	0	0	1	-

```

Conversion completed. One or more failures and/or warnings occurred.
For errors see js2ai.log

```

```
# cat js2ai.log
```

```

sysidcfg:line 6:CONVERSION: DefaultFixed network configuration enabled.
Unable to complete network configuration, replace interface PRIMARY
with the actual interface you wish to configure.
sysidcfg:line 17:WARNING: Oracle Solaris 11 uses roles instead of root user.
An admin user with root role privileges will need to be defined in order to
access the system in multi-user mode. The necessary xml structures have been
added to sc_profile.xml as a comment. Edit sc_profile.xml to perform the
necessary modifications to define the admin user.

```

### 例 3-2 修正された sysidcfg ファイルの変換

次に示す変更された sysidcfg ファイルは、前の例で報告されたエラーに対処します。PRIMARY インタフェース指定はインタフェース名 e1000g で置換されています。

```

timezone=US/Pacific
timeserver=localhost
keyboard=US-English
system_locale=en_US.UTF-8
terminal=vt100
network_interface=e1000g { hostname=host1
    ip_address=192.0.2.4

```

```

netmask=255.255.255.224
protocol_ipv6=yes
default_route=192.0.2.3 }
name_service=DNS{domain_name=example.com
name_server=192.0.2.1,192.0.2.2
search=example.com,example.org}
root_password=rJmv5LUXM10cU
security_policy=none
nfs_domain=dynamic

# js2ai -s5

Name                               Process  Unsupported  Conversion  Validation
Warnings  Errors  Items      Errors      Errors
-----
sysidcfg                2         0         0           0           -
Conversion completed. One or more failures and/or warnings occurred.
For details see /tmp/js2ai.log
# cat js2ai.log
sysidcfg:line 16:WARNING: Oracle Solaris 11 uses roles instead of root user.
An admin user with root role privileges will need to be defined in order to
access the system in multi-user mode. The necessary xml structures have been
added to sc_profile.xml as a comment. Edit sc_profile.xml to perform the
necessary modifications to define the admin user.
sysidcfg:line 6:WARNING: In order to support the direct translation of the
sysidcfg interface 'e1000g', Oracle Solaris 11 neutral link name support will
be disabled. If you wish to use neutral link names, change the interface
name specified in the sysidcfg file to a 'netx' style interface name or edit
the resulting sc_profile.xml file.

```

ネットワークインタフェース primary は Oracle Solaris 10 インタフェース名で置換されていますが、ニュートラルリンク名に関する警告が引き続き表示されます。古いスタイルのリンク名を使用する場合は、必要なアクションはありません。ほとんどの場合、インタフェースは Oracle Solaris 10 の場合と同じようにマップされますが、必ずそうなるという保証はありません。

Oracle Solaris 11 によって使用されるニュートラルリンク名を使用する場合は、sysidcfg ファイルのネットワークインタフェースを net# という形式の名前に変更します。通常、プライマリインタフェースは net0 として定義されます。ユーザー警告が表示される問題に対処するには、前の例の line 6 の注に示したとおりに、sc\_profile.xml ファイルを編集します。

## 詳細な構成変換情報の表示

sysidcfg ファイル変換またはシステム構成プロファイルの検証に関する詳細な情報を表示するには、js2ai コマンドに -v オプションを指定します。-v オプションを指定すると、処理段階が表示されます。

```
# js2ai -sv
```

```
Processing: sysidcfg
Performing conversion on: sysidcfg
Generating SC Profile
Validating sc_profile.xml
```

Name	Warnings	Process Errors	Unsupported Items	Conversion Errors	Validation Errors
-----	-----	-----	-----	-----	-----
sysidcfg	2	0	0	0	0

```
Conversion completed. One or more failures and/or warnings occurred.
For details see /tmp/js2ai.log
```

## 出力システム構成プロファイルの検証

js2ai コマンドを実行した結果得られるシステム構成プロファイルを検証するには、-v オプションを使用します。

```
# js2ai -v -V ./sc_profile.xml
Validating sc_profile.xml
```

Name	Warnings	Process Errors	Unsupported Items	Conversion Errors	Validation Errors
-----	-----	-----	-----	-----	-----
sc_profile	-	-	-	-	0

```
Successfully completed conversion
```

# ◆◆◆ 第 4 章

## Oracle Solaris 11 サーバー上の JumpStart を使用した Oracle Solaris 10 のインストール

---

JumpStart を使用して Oracle Solaris 10 オペレーティングシステムをネットワーク上の SPARC および x86 プラットフォームにインストールすることはできますが、Oracle Solaris 11 OS をインストールすることはできません。ただし、JumpStart インストールサーバーを Oracle Solaris 11 システムにすることはできます。

Oracle Solaris 11 サーバーは、次の 2 つの異なるジョブを実行します。

- Automated Installer を使用して Oracle Solaris 11 OS のインストールを実行します。詳細は、『[Oracle Solaris 11.2 システムのインストール](#)』のパート III「[インストールサーバーを使用したインストール](#)」を参照してください。
- JumpStart を使用して Oracle Solaris 10 OS のインストールを実行します。この章では、Oracle Solaris 11 システム上に JumpStart インストールサーバーを設定する方法を説明します。JumpStart の詳細は、『[Oracle Solaris 10 1/13 Installation Guide: Live Upgrade and Upgrade Planning](#)』を参照してください。

## Oracle Solaris 10 JumpStart サーバーとしての Oracle Solaris 11 システムの設定

次の手順では、Oracle Solaris 11 JumpStart インストールサーバーを作成して、Oracle Solaris 10 OS をクライアントシステム上にインストールする方法を示します。

### ▼ Oracle Solaris 10 JumpStart サーバーとして Oracle Solaris 11 システムを設定する方法

1. Oracle Solaris 11 OS をインストールします。

## 2. 静的 IP アドレスを設定します。

静的 IP アドレスを設定する方法の詳細は、『[Oracle Solaris 11.2 でのネットワークコンポーネントの構成と管理](#)』の「IPv4 インタフェースを構成する方法」および `ipadm(1M)` のマニュアルページに記載されている手順を参照してください。

## 3. `system/boot/network` パッケージを `solaris` パブリッシャーからインストールします。

```
# pkg publisher
PUBLISHER          TYPE    STATUS  URI
solaris            origin  online  http://pkg.oracle.com/solaris11/release/

# pkg install pkg:/system/boot/network
Packages to install: 1
Variants/Facets to change: 3
Create boot environment: No
Create backup boot environment: No
Services to change: 1

DOWNLOAD          PKGS     FILES   XFER (MB)
Completed         1/1      13/13   0.0/0.0

PHASE              ACTIONS
Install Phase     34/34

PHASE              ITEMS
Package State Update Phase  1/1
Image State Update Phase    2/2

PHASE              ITEMS
Reading Existing Index      8/8
Indexing Packages           1/1
```

## 4. 最新の [Oracle Solaris 10 DVD イメージ](#) をダウンロードします。

このイメージは、インストールイメージおよび JumpStart ツールのソースです。JumpStart は、Oracle Solaris 10 OS をインストールするときのみ使用でき、Oracle Solaris 11 OS のインストールには使用できません。

## 5. JumpStart インストールサーバーを設定します。

次の例では、Oracle Solaris 10 1/13 の SPARC 用 DVD イメージを使用しています。

```
# /media/SOL_10_0113_SPARC/Solaris_10/Tools/setup_install_server /export/s10u11_sparc
Verifying target directory...
Calculating the required disk space for the Solaris_10 product
Calculating space required for the installation boot image
Copying the CD image to disk...
Copying Install Boot Image hierarchy...
Copying /boot netboot hierarchy...
Install Server setup complete
```

6. **jumpstart\_sample** ファイルを JumpStart プロファイルの場所にコピーします。

```
# cd /media/SOL_10_0113_SPARC/Solaris_10/Misc/jumpstart_sample
# mkdir -p /export/profiles/s10profile
# cp -pr * /export/profiles/s10profile
```

7. **check** スクリプトを更新します。

/export/profile/s10profile/check スクリプトの 1 行目を次の行で置換します。

```
#!/usr/sunos/bin/sh
```

8. **JumpStart** プロファイルを検証します。

次の例では、簡潔に示すために多くの出力行が省略されています。-p オプションはすべての Oracle Solaris 10 更新リリースで機能するとは限りません。少なくとも Oracle Solaris 10 Update 7 を使用していることを確認してください。

```
# cd /export/profile/s10profile
# ./check -p /export/s10u11_sparc
45 blocks
Validating rules...
Validating profile host_class...
Validating profile zfsrootsimple...
Validating profile net924_sun4c...
Validating profile upgrade...
Validating profile x86-class...
Validating profile any_machine...
The custom JumpStart configuration is ok.
```

9. **udp6** が使用可能でオンラインであることを確認します。

udp6 サービスは、tftp パッケージで配信されます。tftp パッケージがインストールされていることを確認してください。

```
# pkg list tftp
```

tftp パッケージがインストールされていない場合は、次のコマンドを使用してインストールします:

```
# pkg install pkg:/service/network/tftp
```

tftp パッケージをインストールしたら、udp6 サービスが使用可能でオンラインになっていることを確認します。

```
# svcs udp6
STATE          STIME   FMRI
disabled      8:36:55  svc:/network/tftp/udp6:default
# svcadm enable network/tftp/udp6
# svcs udp6
```

```
STATE          STIME    FMRI
online         8:38:28 svc:/network/tftp/udp6:default
```

#### 10. /tftpboot ディレクトリを /etc/netboot ディレクトリとリンクします。

/etc/netboot ディレクトリが存在しない場合は作成します。

JumpStart は /tftpboot をネットワークブート用に使用します。AI は /etc/netboot をネットワークブート用に使用します。この Oracle Solaris 11 サーバーが JumpStart インストールサーバーと AI インストールサーバーの両方の役割を果たす場合は、/tftpboot を /etc/netboot に対するシンボリックリンクにすることで 2 つのツールが共存できるようにする必要があります。

---

**ヒント** - このサーバーを JumpStart インストールサーバーまたは AI インストールサーバーのいずれかとしてのみ使用することを計画している場合、このステップは不要です。ただし、後で計画を変更する場合にネットブートエラーの失敗を回避するためには、いずれにしてもこのステップを実行する必要があります。

---

```
# ls /etc/netboot
/etc/netboot: No such file or directory
# mkdir -m 755 /etc/netboot
# ln -s /etc/netboot /tftpboot
# ls -l /tftpboot
lrwxrwxrwx  1 root    root          12 Sep 14 8:46:51 /tftpboot -> /etc/netboot
```

#### 11. クライアントをインストールします。

クライアントには完全修飾ホスト名を使用してください。この例では、クライアントのホスト名は line2-x4100.example.com です。

```
/export/s10u11_sparc/Solaris_10/Tools/add_install_client -e 8:0:20:fd:f2:18 \
-c line2-x4100.example.com:/export/profiles/s10profile \
-p line2-x4100.example.com:/export/profiles/s10profile line2-t1 sun4u
/export/home is already shared. However, the zfs file system
/export/home must be shared read-only with root access. Use the "zfs
set" command to set the sharenfs property for file system /export/home
as follows:Use ro and either anon=0 or
root=line2-x4100.example.com for /export/home. This must be
fixed and /export/home shared before line2-x4100.exempl.com can boot.
.
.
```

zfs コマンドは変更されているため、次のコマンドを使用して、/export/home ファイルシステムの共有状態を確認し、必要に応じて共有オプションを修正してから、ファイルシステムの共有状態をチェックして変更を検証してください。この例では、anon=0 が設定されていませんでした。

```
# share
home    /export/home    nfs      sec=sys,ro
```

```
share -F nfs -o ro,sec=sys,anon=0 /export/home
# share
home /export/home nfs anon=0,sec=sys,ro
```



# 索引

---

## さ

相違点

JumpStart と AI, 9

## は

比較

JumpStart と AI のインストールタスク, 9

変換

JumpStart ルールおよびプロファイルファイル, 13

## ら

類似点

JumpStart と AI, 9

ルートディスク

決定, 23

## A

AI 参照 Automated Installer (AI)

AI 条件の指令

JumpStart キーワードとの比較, 13

AI プロファイル

JumpStart ファイルを変換, 13

AI マニフェスト

JumpStart ファイルを変換, 13

AI マニフェストの指令

JumpStart キーワードとの比較, 15

any JumpStart ルールファイルキーワード, 14

any デバイスの変換

js2ai コマンドを使用, 24

arch AI 選択条件, 14

arch JumpStart ルールファイルキーワード, 14

archive\_location JumpStart プロファイルファイル  
キーワード, 16

Automated Installer (AI)

JumpStart との比較, 9

## B

backup\_media JumpStart プロファイルファイルキー  
ワード, 16

boot\_device JumpStart プロファイルファイルキー  
ワード, 16

bootenv JumpStart プロファイルファイルキーワード,  
16

## C

client\_arch JumpStart プロファイルファイルキー  
ワード, 16

client\_root JumpStart プロファイルファイルキー  
ワード, 16

client\_swap JumpStart プロファイルファイルキー  
ワード, 16

cluster JumpStart プロファイルファイルキーワード,  
16

cpu AI 選択条件, 14

## D

disksize JumpStart ルールファイルキーワード, 14

domainname JumpStart ルールファイルキーワード, 14

dontuse JumpStart プロファイルファイルキーワード,  
17

## F

fdisk JumpStart プロファイルファイルキーワード, 17

filesystems JumpStart プロファイルファイルキーワード, 17  
forced\_deployment JumpStart プロファイルファイルキーワード, 19

## G

geo JumpStart プロファイルファイルキーワード, 19

## H

hostaddress JumpStart ルールファイルキーワード, 14  
hostname JumpStart ルールファイルキーワード, 14

## I

install\_type JumpStart プロファイルファイルキーワード, 19  
installed JumpStart ルールファイルキーワード, 14  
ipv4 AI 選択条件, 14, 15

## J

js2ai コマンド  
  any デバイスを変換する, 24  
  JumpStart ファイルの変換, 22  
  ZFS ルートプールの決定, 24  
  ルートディスクの決定, 23  
JumpStart  
  AI との比較, 9  
JumpStart から AI への移行  
  概要, 10  
JumpStart プロファイルファイル  
  AI でサポートされていないキーワード, 15  
  AI ファイルへの変換, 13  
  AI マニフェストの指令と比較した JumpStart キーワード, 15  
JumpStart ルールファイル  
  AI 条件の指令と比較した JumpStart キーワード, 13  
  AI でサポートされていないキーワード, 13  
  AI ファイルへの変換, 13

## K

karch JumpStart ルールファイルキーワード, 14

## L

layout\_constraint JumpStart プロファイルファイルキーワード, 19  
local\_customization JumpStart プロファイルファイルキーワード, 19  
locale JumpStart プロファイルファイルキーワード, 19

## M

mac AI 選択条件, 14  
mem AI 選択条件, 14  
memsize JumpStart ルールファイルキーワード, 14  
metadb JumpStart プロファイルファイルキーワード, 19  
model JumpStart ルールファイルキーワード, 15

## N

network AI 選択条件, 15  
network JumpStart ルールファイルキーワード, 15  
no\_content\_check JumpStart プロファイルファイルキーワード, 19  
no\_master\_check JumpStart プロファイルファイルキーワード, 19  
num\_clients JumpStart プロファイルファイルキーワード, 19

## O

osname JumpStart ルールファイルキーワード, 15

## P

package JumpStart プロファイルファイルキーワード, 19  
partitioning JumpStart プロファイルファイルキーワード, 20

---

patch JumpStart プロファイルファイルキーワード, 20  
platform AI 選択条件, 15  
pool JumpStart プロファイルファイルキーワード, 20  
probe JumpStart ルールファイルキーワード, 15

## R

root\_device JumpStart プロファイルファイルキーワード, 21

## S

system\_type JumpStart プロファイルファイルキーワード, 22

## T

totaldisk JumpStart ルールファイルキーワード, 15

## U

usedisk JumpStart プロファイルファイルキーワード, 22

## Z

ZFS ルートプール  
決定, 24

