

Oracle® Solaris 10 JumpStart에서 Oracle Solaris 11.1 자동 설치 프로그램으로 전환

Copyright © 2011, 2012, Oracle and/or its affiliates. All rights reserved.

본 소프트웨어와 관련 문서는 사용 제한 및 기밀 유지 규정을 포함하는 라이선스 계약서에 의거해 제공되며, 지적 재산법에 의해 보호됩니다. 라이선스 계약서 상에 명시적으로 허용되어 있는 경우나 법규에 의해 허용된 경우를 제외하고, 어떠한 부분도 복사, 재생, 번역, 방송, 수정, 라이선스, 전송, 배포, 진열, 실행, 발행, 또는 전시될 수 없습니다. 본 소프트웨어를 리버스 엔지니어링, 디스어셈블리 또는 디컴파일하는 것은 상호 운용에 대한 법규에 의해 명시된 경우를 제외하고는 금지되어 있습니다.

이 안의 내용은 사전 공지 없이 변경될 수 있으며 오류가 존재하지 않음을 보증하지 않습니다. 만일 오류를 발견하면 서면으로 통지해 주시기 바랍니다.

만일 본 소프트웨어나 관련 문서를 미국 정부나 또는 미국 정부를 대신하여 라이선스한 개인이나 법인에게 배송하는 경우, 다음 공지 사항이 적용됩니다.

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.

본 소프트웨어 혹은 하드웨어는 다양한 정보 관리 애플리케이션의 일반적인 사용을 목적으로 개발되었습니다. 본 소프트웨어 혹은 하드웨어는 개인적인 상해를 초래할 수 있는 애플리케이션을 포함한 본질적으로 위험한 애플리케이션에서 사용할 목적으로 개발되거나 그 용도로 사용될 수 없습니다. 만일 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서 사용할 경우, 라이선스 사용자는 해당 애플리케이션의 안전한 사용을 위해 모든 적절한 비상-안전, 백업, 대비 및 기타 조치를 반드시 취해야 합니다. Oracle Corporation과 그 회사는 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서의 사용으로 인해 발생하는 어떠한 손해에 대해서도 책임지지 않습니다.

Oracle과 Java는 Oracle Corporation 및/또는 그 자회사의 등록 상표입니다. 기타의 명칭들은 각 해당 명칭을 소유한 회사의 상표일 수 있습니다.

Intel 및 Intel Xeon은 Intel Corporation의 상표 내지는 등록 상표입니다. SPARC 상표 일체는 라이선스에 의거하여 사용되며 SPARC International, Inc.의 상표 내지는 등록 상표입니다. AMD, Opteron, AMD 로고, 및 AMD Opteron 로고는 Advanced Micro Devices의 상표 내지는 등록 상표입니다. UNIX는 The Open Group의 등록 상표입니다.

본 소프트웨어 혹은 하드웨어와 관련 문서(설명서)는 제 3자로부터 제공되는 콘텐츠, 제품 및 서비스에 접속할 수 있거나 정보를 제공합니다. Oracle Corporation과 그 자회사는 제 3자의 콘텐츠, 제품 및 서비스와 관련하여 어떠한 책임도 지지 않으며 명시적으로 모든 보증에 대해서도 책임을 지지 않습니다. Oracle Corporation과 그 자회사는 제 3자의 콘텐츠, 제품 및 서비스에 접속하거나 사용으로 인해 초래되는 어떠한 손실, 비용 또는 손해에 대해 어떠한 책임도 지지 않습니다.

목차

| | |
|---|-----------|
| 머리말 | 7 |
| 1 JumpStart와 자동 설치 프로그램 비교 | 11 |
| JumpStart와 AI의 비슷한 점과 차이점 | 11 |
| JumpStart에서 AI로 마이그레이션 정보 | 12 |
| 2 규칙 및 프로파일 파일 변환 | 15 |
| 규칙 키워드와 조건 지시어 비교 | 15 |
| 프로파일 키워드와 AI 매니페스트 지시어 비교 | 18 |
| js2ai를 사용하여 JumpStart 규칙 및 프로파일을 AI 조건 및 매니페스트로 변환 | 24 |
| js2ai가 JumpStart 장치 지정을 변환하는 방식 | 25 |
| 소프트웨어 패키지 변환 | 26 |
| js2ai를 사용한 규칙 및 프로파일 변환 예 | 29 |
| 추가 프로파일 변환 정보 표시 | 37 |
| 출력 AI 매니페스트 검증 | 37 |
| 3 구성 파일 변환 | 39 |
| sysidcfg 파일 키워드와 시스템 구성 프로파일 지시어 비교 | 39 |
| js2ai를 사용하여 sysidcfg 파일을 시스템 구성 프로파일로 변환 | 41 |
| js2ai를 사용한 sysidcfg 변환 예 | 42 |
| 추가 구성 변환 정보 표시 | 44 |
| 출력 시스템 구성 프로파일 검증 | 44 |
| 4 JumpStart를 사용하여 Oracle Solaris 11 서버에 Oracle Solaris 10 설치 | 45 |
| Oracle Solaris 11 시스템을 Oracle Solaris 10 JumpStart 서버로 설정 | 45 |
| 알려진 문제에 대한 임시 해결책 | 48 |
| rm_install_client 스크립트 문제 | 48 |

check 스크립트 문제 48

표

| | | |
|-------|--|----|
| 표 1-1 | JumpStart와 AI 단계 비교 | 11 |
| 표 2-1 | JumpStart 규칙 파일 키워드와 AI 조건 지시어 비교 | 15 |
| 표 2-2 | JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 | 18 |
| 표 3-1 | sysidcfg 파일 키워드와 구성 프로파일 지시어 비교 | 39 |

머리말

Oracle Solaris 10 JumpStart에서 Oracle Solaris 11.1 자동 설치 프로그램으로 전환은 JumpStart에서 AI(자동 설치 프로그램)로 마이그레이션하는 데 유용한 정보를 제공합니다. 두 프로그램 모두 Oracle Solaris의 자동 설치 기능을 제공합니다.

본 설명서에서 설명하는 절차와 도구는 다음과 같습니다.

- JumpStart와 AI의 차이점
- JumpStart 규칙 및 프로파일 키워드를 AI 조건 및 매니페스트 키워드로 변환하는 방법
- sysidcfg 파일을 AI 시스템 구성 프로파일로 변환하는 방법
- js2ai 변환 유틸리티를 사용하는 방법
- Oracle Solaris 10과 Oracle Solaris 11 설치를 모두 제공하도록 하나의 서버를 설정하는 방법

이 설명서의 대상

본 설명서는 AI를 사용하여 Oracle Solaris 11 OS(운영 체제) 자동 설치를 수행하도록 변환하려는 JumpStart 사용자를 대상으로 합니다.

이 설명서의 구성

본 설명서는 다음과 같은 장으로 구성됩니다.

- 1 장, “JumpStart와 자동 설치 프로그램 비교”에서는 각각 JumpStart 및 AI를 사용하여 작업을 수행하는 방법을 알려주는 설치 작업 표를 제공합니다.
- 2 장, “규칙 및 프로파일 파일 변환”에서는 JumpStart 규칙 및 프로파일 키워드와 이에 해당하는 AI 조건 및 매니페스트 지시어에 대한 표를 제공합니다. 이 장에서는 js2ai 유틸리티를 사용한 변환 예제도 보여줍니다.
- 3 장, “구성 파일 변환”에서는 sysidcfg 파일 키워드와 이에 해당하는 AI 시스템 구성 프로파일 지정 사항에 대한 표를 제공하며, js2ai 유틸리티를 사용한 변환 예제를 보여줍니다.
- 4 장, “JumpStart를 사용하여 Oracle Solaris 11 서버에 Oracle Solaris 10 설치”에서는 하나의 Oracle Solaris 11 서버를 JumpStart 설치 서버와 AI 설치 서버로 사용하는 방법에 대해 설명합니다.

관련 정보

Oracle Solaris 10 8/11 설치 설명서: 사용자 정의 JumpStart 및 고급 설치에서는 JumpStart 사용법에 대해 설명합니다.

Oracle Solaris 11.1 시스템에서는 AI 등을 사용하여 Oracle Solaris 11.1 OS를 설치하는 여러 가지 방법을 설명합니다.

Oracle Solaris 11.1에서 서비스 및 결합 관리의 1 장, “서비스 관리(개요)”에서는 Oracle Solaris SMF(서비스 관리 기능) 기능에 대해 설명합니다. SMF 프로파일을 사용하여 시스템을 구성할 수 있습니다.

pkg(5) 매뉴얼 페이지에서는 설치할 소프트웨어 패키지를 저장하고 검색할 수 있도록 해주는 Oracle Solaris IPS(Image Packaging System) 기능에 대해 설명합니다. **pkg(1)** 매뉴얼 페이지에서는 IPS 패키지를 설치하는 방법에 대해 설명합니다.

Oracle Solaris 11.1 시스템 관리 방법에 대한 자세한 내용은 Oracle Solaris 11.1 System Administration 설명서를 참조하십시오.

Oracle Support에 액세스

Oracle 고객은 My Oracle Support를 통해 온라인 지원에 액세스할 수 있습니다. 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>를 참조하거나, 청각 장애가 있는 경우 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>를 방문하십시오.

활자체 규약

다음 표는 이 설명서에서 사용되는 활자체 규약에 대해 설명합니다.

표 P-1 활자체 규약

| 활자체 | 설명 | 예 |
|-----------|---------------------------------------|--|
| AaBbCc123 | 명령, 파일, 디렉토리 이름 및 컴퓨터 화면에 출력되는 내용입니다. | .login 파일을 편집하십시오. 모든 파일 목록을 보려면 <code>ls -a</code> 명령을 사용하십시오. machine_name% you have mail. |
| AaBbCc123 | 사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 대조됩니다. | machine_name% su Password: |
| AaBbCc123 | 위치 표시자: 실제 이름이나 값으로 바꿉니다. | <code>rm filename</code> 명령을 사용하여 파일을 제거합니다. |

표 P-1 활자체 규약 (계속)

| 활자체 | 설명 | 예 |
|-----------|-----------------------------|--|
| AaBbCc123 | 설명서 제목, 새 용어, 강조 표시할 용어입니다. | <p>사용자 설명서의 6장을 읽으십시오.</p> <p>캐시는 로컬로 저장된 복사본입니다.</p> <p>파일을 저장하면 안 됩니다.</p> <p>주: 일부 강조된 항목은 온라인에서 굵은체로 나타납니다.</p> |

명령 예의 셸 프롬프트

다음 표에는 Oracle Solaris OS에 포함된 셸의 기본 UNIX 시스템 프롬프트 및 슈퍼유저 프롬프트가 나와 있습니다. 명령 예제에 표시된 기본 시스템 프롬프트는 Oracle Solaris 릴리스에 따라 다릅니다.

표 P-2 셸 프롬프트

| 셸 | 프롬프트 |
|---------------------------------|---------------|
| Bash 셸, Korn 셸 및 Bourne 셸 | \$ |
| 슈퍼유저용 Bash 셸, Korn 셸 및 Bourne 셸 | # |
| C 셸 | machine_name% |
| 슈퍼유저용 C 셸 | machine_name# |

JumpStart와 자동 설치 프로그램 비교

이 장에서는 다음 정보를 제공합니다.

- JumpStart와 AI(자동 설치 프로그램)의 비슷한 점과 차이점
- JumpStart에서 AI로 마이그레이션하는 데 도움이 되는 참조 정보

JumpStart와 AI의 비슷한 점과 차이점

JumpStart와 AI 모두 네트워크에 있는 여러 시스템의 핸드 프리 설치를 제공합니다. 클라이언트가 네트워크 부트되며, 클라이언트가 부트되면 설치 프로그램이 인계합니다.

JumpStart는 Oracle Solaris 10 OS 및 Oracle Solaris OS의 이전 버전을 설치합니다. AI는 Oracle Solaris 11 OS 및 업데이트 릴리스를 설치합니다.

JumpStart와 AI는 다음과 같은 특징을 공유합니다.

- 설치 서버에 시스템 구성을 저장함으로써 여러 클라이언트의 핸드 프리 네트워크 설치를 제공합니다.
- 한 번의 자동 설치로 다른 클라이언트에 여러 종류의 설치를 제공합니다.
- x86 클라이언트와 SPARC 클라이언트를 모두 설치합니다.

표 1-1 JumpStart와 AI 단계 비교

| 단계 | JumpStart | AI |
|--------------|-------------------------------------|--------------------------------------|
| 설치 서버 설정 | setup_install_server(1M) 명령을 사용합니다. | installadm create-service 명령을 사용합니다. |
| 설치에 클라이언트 추가 | add_install_client(1M) 명령을 사용합니다. | installadm create-client 명령을 사용합니다. |
| 파생된 프로파일 만들기 | 시작 스크립트를 사용합니다. | 파생된 매니페스트 방식을 사용합니다. |

표 1-1 JumpStart와 AI 단계 비교 (계속)

| 단계 | JumpStart | AI |
|-----------------|-------------------------------------|---|
| 설치 명령 지정 | 프로파일 파일을 사용합니다. | AI 매니페스트 파일을 사용합니다. |
| 클라이언트 프로비전 | DVD를 사용합니다. | 인터넷 또는 로컬 네트워크를 통해 패키지 저장소를 사용합니다. |
| 클라이언트 사용자 정의 지정 | 규칙 파일을 사용하여 클라이언트를 프로파일 파일과 연관시킵니다. | installadm 명령의 create-manifest, create-profile 또는 set-criteria 하위 명령을 사용하여 클라이언트를 AI 매니페스트 및 시스템 구성 프로파일과 연관시킵니다. |
| 클라이언트 구성 지정 | 종료 스크립트 및 sysidcfg(4) 파일을 사용합니다. | SMF 시스템 구성 프로파일 파일을 사용합니다. 처음 부트할 때 한번 실행되는 SMF 서비스를 통해 실행되는 스크립트를 사용합니다. |

JumpStart에서 AI로 마이그레이션 정보

JumpStart 규칙, 프로파일 및 구성 파일을 AI 조건, AI 매니페스트 및 SMF 시스템 구성 프로파일로 변환

js2ai 유틸리티를 가져옵니다.

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

JumpStart 데이터를 AI 데이터로 변환하려면 다음 지침을 따르십시오.

- js2ai(1M)
- 2 장, “규칙 및 프로파일 파일 변환”
- 3 장, “구성 파일 변환”

하나의 서버를 JumpStart 설치 서버와 AI 설치 서버로 사용

4 장, “JumpStart를 사용하여 Oracle Solaris 11 서버에 Oracle Solaris 10 설치”

AI 클라이언트 프로비저닝 매니페스트 동적 파생

Oracle Solaris 11.1 시스템의 “클라이언트 설치 타임에 AI 매니페스트 만들기”를 참조하십시오. JumpStart 시작 스크립트를 사용하여 설치 프로그램에 전달되는 설치 매개변수를 동적으로 조작할 수 있습니다. AI를 사용하여 클라이언트 설치 시 클라이언트 속성을 쿼리하고 해당 클라이언트에 알맞게 사용자 정의된 프로비저닝 매니페스트를 동적으로 파생시킬 수 있습니다. 환경 변수에는 클라이언트의 하드웨어 속성이 지정되며, 이들 대부분은 JumpStart 시작 스크립트에서 사용된 환경 변수와 동일합니다.

AI 설치를 위해 소프트웨어 패키지 저장소 액세스

인터넷(pkg.oracle.com)을 통해 Oracle Solaris 11 패키지 저장소를 사용합니다.

패키지의 로컬 복사본을 만듭니다(**Oracle Solaris 11.1 패키지 저장소 복사 및 만들기**).

시스템 구성 지침 제공

SMF 프로파일을 만드는 방법은 **Oracle Solaris 11.1 시스템**의 11 장, “클라이언트 시스템 구성”을 참조하십시오.

AI 매니페스트 및 SMF 프로파일을 만들어 AI 클라이언트 설치의 일부로 비전역 영역을 설치하는 방법은 **Oracle Solaris 11.1 시스템**의 12 장, “영역 설치 및 구성”을 참조하십시오.

처음 부트할 때 한 번 실행되며 사용자 정의 스크립트를 실행하는 SMF 서비스 만들기 **Oracle Solaris 11.1 시스템**의 13 장, “처음 부트 중 사용자 정의 스크립트 실행”을 참조하십시오.

하나의 설치 파일에 원하는 설치 만들기

사용자 정의 설치 이미지를 만드는 방법은 **사용자 정의 Oracle Solaris 11.1 설치 이미지 만들기**를 참조하십시오.

규칙 및 프로파일 파일 변환

이 장에서는 JumpStart 규칙 및 프로파일 파일을 AI 조건 파일 및 AI 매니페스트로 변환하는 방법을 보여줍니다. 대부분의 변환은 js2ai 유틸리티를 사용하여 수행할 수 있습니다. js2ai 유틸리티에 대한 자세한 내용은 [js2ai\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

AI 매니페스트는 디스크 레이아웃 및 설치할 소프트웨어 패키지와 같은 시스템 프로비저닝을 지정하는 XML 파일입니다. AI 매니페스트에 대한 자세한 내용은 [Oracle Solaris 11.1 시스템의 10 장, “클라이언트 시스템 프로비전”](#) 및 [ai_manifest\(4\)](#) 매뉴얼 페이지를 참조하십시오.

AI는 클라이언트 조건을 사용하여 어떤 클라이언트 시스템이 어떤 매니페스트를 사용하여 설치를 완료하는지를 지정합니다. AI 매니페스트가 AI 설치 서비스에 추가되면 명령줄이나 파일에 조건을 지정할 수 있습니다. 이 장에서는 js2ai 도구를 사용하여 JumpStart rules 파일을 AI 조건 파일로 변환하는 방법을 보여줍니다. AI 클라이언트 조건 지정에 대한 자세한 내용은 [Oracle Solaris 11.1 시스템의 9 장, “설치 사용자 정의”](#)를 참조하십시오.

규칙 키워드와 조건 지시어 비교

다음 표는 JumpStart 규칙 키워드를 AI 조건 지시어와 비교합니다. AI는 이러한 조건을 사용하여 올바른 AI 매니페스트 파일 또는 시스템 구성 프로파일을 특정 클라이언트에 적용합니다.

표 2-1 JumpStart 규칙 파일 키워드와 AI 조건 지시어 비교

| JumpStart 규칙 파일 키워드 | AI 조건 파일 지시어 |
|---------------------|--|
| any | 지원되지 않습니다. 선택 조건과 일치하지 않는 클라이언트 시스템의 경우 AI 설치 서비스가 기본 AI 매니페스트를 제공합니다. |

표 2-1 JumpStart 규칙 파일 키워드와 AI 조건 지시어 비교 (계속)

| JumpStart 규칙 파일 키워드 | AI 조건 파일 지시어 |
|---------------------|--|
| arch | cpu 명령 옵션: -c cpu=sparc 조건 파일: <pre><ai_criteria name="cpu"> <value>sparc</value> </ai_criteria></pre> |
| disksize | 지원되지 않습니다. 해당 디스크가 필요한 최소 크기를 충족할 경우 AI는 AI 매니페스트에 지정된 대상에 설치합니다. |
| domainname | 지원되지 않습니다. |
| hostaddress | ipv4 명령 옵션: -c ipv4=10.6.68.127 조건 파일: <pre><ai_criteria name="ipv4"> <value>10.6.68.127</value> </ai_criteria></pre> |
| hostname | 지원되지 않습니다. AI에서 호스트를 고유하게 식별하려면 위에 표시된 것과 같이 IP 주소를 사용하거나, MAC 주소를 사용하십시오. 명령 옵션: -c mac=0:3:ba:33:9d:b6 조건 파일: <pre><ai_criteria name="mac"> <value>0:3:ba:33:9d:b6</value> </ai_criteria></pre> |
| installed | 지원되지 않습니다. 해당 디스크가 필요한 최소 크기를 충족할 경우 AI는 AI 매니페스트에 지정된 대상에 설치합니다. |
| karch | arch 명령 옵션: -c arch=i86pc 조건 파일: <pre><ai_criteria name="arch"> <value>i86pc</value> </ai_criteria></pre> |

표 2-1 JumpStart 규칙 파일 키워드와 AI 조건 지시어 비교 (계속)

| JumpStart 규칙 파일 키워드 | AI 조건 파일 지시어 |
|---------------------|---|
| memsize | <p>mem</p> <p>명령 옵션: -c mem=2048</p> <p>조건 파일:</p> <pre><ai_criteria name="mem"> <value>2048</value> </ai_criteria></pre> |
| model | <p>platform</p> <p>명령 옵션: -c platform=SUNW,Sun-Fire-T200</p> <p>조건 파일:</p> <pre><ai_criteria name="platform"> <value>SUNW,Sun-Fire-T200</value> </ai_criteria></pre> |
| network | <p>network, 범위를 포함하는 network 또는 범위를 포함하는 ipv4를 사용합니다.</p> <p>명령 옵션, 단일 network 값: -c network="10.0.0.0"</p> <p>조건 파일, 단일 network 값:</p> <pre><ai_criteria name="network"> <value>10.0.0.0</value> </ai_criteria></pre> <p>명령 옵션, ipv4 범위: -c ipv4=10.0.0.1-10.0.0.64</p> <p>조건 파일, ipv4 범위:</p> <pre><ai_criteria name="ipv4"> <range>10.0.0.1 10.0.0.64</range> </ai_criteria></pre> |
| osname | 지원되지 않습니다. |
| probe | 지원되지 않습니다. |
| totaldisk | 지원되지 않습니다. 해당 디스크가 필요한 최소 크기를 충족할 경우 AI는 AI 매니페스트에 지정된 대상에 설치합니다. |

프로파일 키워드와 AI 매니페스트 지시어 비교

다음 표는 JumpStart 프로파일 키워드를 AI 매니페스트 지시어와 비교합니다. AI는 XML 매니페스트 파일을 사용하여 클라이언트 설치를 정의합니다. AI 매니페스트에 대한 자세한 내용은 **Oracle Solaris 11.1 시스템의 10 장**, “클라이언트 시스템 프로비전” 및 **ai_manifest(4)** 매뉴얼 페이지를 참조하십시오.

any로 지정된 장치와 같이, 클라이언트 설치 프로세스가 시작될 때까지 알려지지 않는 값을 지정하려면 파생된 매니페스트 스크립트를 사용하십시오. 파생된 매니페스트 스크립트를 사용하면 디스크 크기를 기준으로 스왑 크기를 지정하거나 사용 가능한 디스크를 기준으로 미러링을 지정할 수 있습니다. 파생된 매니페스트 스크립트에 대한 자세한 내용은 **Oracle Solaris 11.1 시스템의 “클라이언트 설치 타임에 AI 매니페스트 만들기”**를 참조하십시오.

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|--|--|
| archive_location | 지원되지 않습니다. AI가 플래시 아카이브에서 설치하지 않습니다. |
| backup_media | 지원되지 않습니다. backup_media 키워드는 install_type의 upgrade 옵션과 함께 사용됩니다. AI는 업그레이드 설치 유형은 지원하지 않고, 초기 설치만 지원합니다. |
| boot_device device boot_device c1t0d0 | <pre><target> <disk whole_disk="true"> <disk_name name="c1t0d0" name_type="ctd"/> </disk> <logical nodump="true" noswap="false"/> </target></pre> |
| boot_device device eeprom boot_device c0t0d0s0 update | 위와 동일합니다. 이 키워드의 두 번째 토큰 값(SPARC 시스템의 경우 update, x86 시스템의 경우 preserve)은 AI에서 지원되지 않습니다. AI에서 SPARC 시스템의 EEPROM은 항상 지정된 대상 장치로 업데이트되어 설치된 시스템이 자동으로 해당 장치에서 부트되도록 합니다. x86 시스템에서는 펌웨어가 업데이트되지 않습니다. |
| bootenv | 지원되지 않습니다. AI는 부트 환경을 만들고 해당 부트 환경에 Oracle Solaris 11 OS를 설치합니다. |
| client_arch | 지원되지 않습니다. 설치할 클라이언트 구조는 AI 매니페스트가 아닌 AI 설치 서비스에 의해 정의됩니다. |
| client_root | 지원되지 않습니다. 대상 디스크의 유형, 공급 업체 및 크기를 지정할 수 있습니다. 대상 디스크 루트 공간의 양은 지정할 수 없습니다. ai_manifest(4) 매뉴얼 페이지를 참조하십시오. |
| client_swap | 지원되지 않습니다. 기본적으로 AI는 루트 폴에 스왑 볼륨을 만듭니다. 스왑 슬라이스를 지정할 수 있습니다. ai_manifest(4) 매뉴얼 페이지를 참조하십시오. |

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 (계속)

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|---|---|
| <code>cluster cluster-name</code> | <p>지원되지 않습니다. <code>cluster</code> 및 <code>package</code> 지정을 처리하는 방법은 26 페이지 “소프트웨어 패키지 변환”을 참조하십시오.</p> <p>Oracle Solaris 11 OS에서는 <code>pkg(5)</code> 그룹 패키지를 사용합니다. 그룹 패키지는 다른 패키지가 매니페스트에 지정된 것처럼 지정됩니다. 기본 AI 매니페스트에는 표준 Oracle Solaris 11 설치에 필요한 패키지가 포함되어 있습니다. 이 패키지 목록은 사용자 정의할 수 있습니다.</p> |
| <code>cluster cluster-name delete</code> | <p>지원되지 않습니다. <code>delete</code> 스위치는 <code>install_type</code>의 <code>upgrade</code> 옵션과 함께 사용됩니다. AI는 업그레이드 설치 유형은 지원하지 않고, 초기 설치만 지원합니다.</p> |
| <code>dontuse</code> | <p>지원되지 않습니다.</p> |
| <code>fdisk disk_name type size</code> <code>fdisk c0t3d0 solaris all</code> | <pre><target> <disk> <disk_name name="c0t3d0" name_type="ctd"/> <partition action="create" name="1" part_type="191"/> </disk> <logical nodump="true" noswap="false"/> </target></pre> <p>AI에서 지원하는 디스크 및 분할 영역 속성의 전체 목록은 <code>ai_manifest(4)</code> 매뉴얼 페이지를 참조하십시오.</p> <p>js2ai에 의한 변환:</p> <p>js2ai 변환의 경우, <code>disk_name</code>의 값은 장치여야 합니다. <code>all</code> 장치는 지원되지 않습니다. <code>fdisk type</code>은 <code>solaris</code>여야 합니다. 0 크기 또는 <code>delete</code>는 지원되지 않습니다. 분할이 <code>default</code>이고 <code>rootdisk</code>가 설정되지 않은 경우, js2ai가 먼저 루트 디스크로 발견된 <code>fdisk solaris</code> 분할 영역을 설정합니다.</p> |
| <code>filesystem</code> | <p>UFS 파일 시스템은 지원되지 않습니다. AI는 ZFS 파일 시스템을 설치합니다.</p> <p>js2ai에 의한 변환:</p> <p>루트에 사용할 장치를 확인할 방법이 없는 경우 <code>filesystem</code> 라인에서 / 마운트 지점을 포함하는 장치가 루트 풀에 사용됩니다.</p> <p>지정된 마운트 지점이 / 또는 <code>swap</code>인 경우 로컬 파일 시스템과 미러링된 파일 시스템이 지원됩니다.</p> <p>마운트 지점이 / 또는 <code>swap</code>이 아닌 경우 해당 라인이 기록된 다음 무시됩니다. JumpStart <code>fsoptions</code>는 지원되지 않습니다.</p> <p>크기 검증은 수행되지 않습니다. 이 매니페스트로 성공적인 설치를 수행하기 위해서는 생성된 AI 매니페스트에 지정된 크기를 조정해야 할 수 있습니다.</p> |

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 (계속)

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|--------------------------|---|
| filesys clt0d0s0 10000 / | <p>다음 일부 AI 매니페스트는 x86 플랫폼용입니다.</p> <pre><target> <disk> <disk_name name="clt0d0" 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="10000mb"/> </slice> </partition> </disk> <logical nodump="true" noswap="false"> <zpool is_root="true" name="rpool"> <vdev name="rpool_vdev" redundancy="none"/> </zpool> </logical> </target></pre> <p>js2ai에 의한 변환:</p> <p>js2ai 도구는 루트 파일 시스템(/) 및 swap의 변환만 지원합니다.</p> |

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 (계속)

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|---|---|
| filesys mirror:rpool c6t0d0s0 c6t1d0s0 60048 / | 다음 일부 AI 매니페스트는 x86 플랫폼용입니다. SPARC 플랫폼의 경우 partition 요소가 지정되지 않습니다. <pre> <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 nodump="true" noswap="false"> <zpool is_root="true" name="rpool"> <vdev name="rpool_vdev" redundancy="mirror"/> </zpool> </logical> </target> </pre> |
| filesys server:path | 지원되지 않습니다. AI는 원격 파일 시스템을 지원하지 않습니다. |
| forced_deployment | 지원되지 않습니다. AI가 플래시 아카이브에서 설치하지 않습니다. |
| geo | 지원되지 않습니다. AI에서는 언어 지원에 대한 지역이 시스템 구성 프로파일에 지정되어 있습니다. Oracle Solaris 11.1 시스템의 11 장 , “클라이언트 시스템 구성”을 참조하십시오. |
| install_type | AI는 initial_install만 지원합니다. |
| layout_constraint | 지원되지 않습니다. AI는 initial_install만 지원합니다. |
| local_customization | 지원되지 않습니다. AI가 플래시 아카이브에서 설치하지 않습니다. |
| locale | 지원되지 않습니다. AI에서는 로케일 지원이 시스템 구성 프로파일에 지정되어 있습니다. Oracle Solaris 11.1 시스템의 11 장 , “클라이언트 시스템 구성”을 참조하십시오. |
| metadb | 지원되지 않습니다. |

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 (계속)

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|--|---|
| no_content_check | 지원되지 않습니다. AI가 플래시 아카이브에서 설치하지 않습니다. |
| no_master_check | 지원되지 않습니다. AI가 플래시 아카이브에서 설치하지 않습니다. |
| num_clients | 지원되지 않습니다. |
| package SUNWpampkcs11 package SUNWpampkcs11 add | <pre><software type="IPS"> <software_data action="install"> <name>pkg:/SUNWpampkcs11</name> </software_data> </software></pre> <p>패키지 이름이 유효한지 확인하는 방법은 26 페이지 “소프트웨어 패키지 변환”을 참조하십시오.</p> |
| package SUNWpampkcs11 add nfs golden:/packages/Solaris_10 | <p>AI 매니페스트에서는 모든 소프트웨어를 IPS 패키지 저장소에서 검색합니다. IPS 저장소 위치는 HTTP 또는 파일일 수 있습니다. 여러 개의 IPS 저장소를 소프트웨어 소스로 지정할 수 있습니다. 게시자를 지정하는 방법은 ai_manifest(4) 매뉴얼 페이지를 참조하십시오.</p> <p>js2ai에 의한 변환:</p> <p>js2ai 도구는 retrieval_type location 인수를 무시합니다.</p> |
| package SUNWpampkcs11 delete | <pre><software type="IPS"> <software_data action="uninstall"> <name>pkg:/SUNWpampkcs11</name> </software_data> </software></pre> <p>AI는 초기 설치만 지원하므로 AI 매니페스트를 사용하여 패키지를 삭제할 필요가 없습니다.</p> |
| partitioning default partitioning explicit | <p>대상 설치를 지정하지 않은 경우 AI가 기본 설치 장치 및 기본 분할을 결정합니다. 사용자 정의 분할을 지정하려면 ai_manifest(4) 매뉴얼 페이지를 참조하십시오.</p> <p>js2ai에 의한 변환:</p> <p>js2ai의 경우, 분할은 default 또는 explicit여야 합니다. explicit의 경우, swap 및 /만 지원됩니다</p> |
| partitioning existing | 지원되지 않습니다. js2ai의 경우, 분할은 default 또는 explicit여야 합니다. |
| patch | <p>지원되지 않습니다. AI는 initial_install만 지원합니다.</p> <p>시스템 또는 특정 패키지를 업데이트하려면 설치된 시스템에서 pkg update 명령을 사용합니다.</p> |

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 (계속)

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|--------------------------------------|--|
| pool newpool auto auto auto c1t0d0s0 | <pre><target> <disk> <disk_name name="c1t0d0" 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 nodump="false" noswap="false"> <zpool is_root="true" name="newpool"> <vdev name="rpool_vdev" redundancy="none"/> </zpool> </logical> </target></pre> <p>js2ai에 의한 변환:</p> <p>프로파일에 풀이 지정된 경우 js2ai는 지정된 장치를 사용하여 ZFS 루트 풀을 만듭니다. js2ai가 ZFS 루트 풀에 사용할 장치를 결정하는 경우 pool 키워드가 다른 모든 키워드를 대체합니다.</p> <p>js2ai 도구는 풀 크기, 스왑 크기 또는 덤프 크기에 대한 검증을 수행하지 않습니다. 이 매니페스트로 성공적인 설치를 수행하기 위해서는 생성된 AI 매니페스트에서 이러한 크기를 조정해야 할 수 있습니다.</p> |
| pool newpool auto auto auto any | <p>js2ai의 경우, 물리적 장치 이름 대신 any를 지정하면 any 매개변수를 포함하는 지정 사항 앞에 장치 정보를 제공해야 합니다. 예를 들어 이 pool 지정 앞에 root_device 또는 usedisk 지정을 제공할 수 있습니다. 예를 보려면 34 페이지 “mirrorpool.profile 오류 수정”을 참조하십시오.</p> |
| root_device c1t0d0s0 | <pre><target> <disk> <disk_name name="c1t0d0" 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"/> </partition> </disk> <logical nodump="true" noswap="false"> <zpool is_root="true" name="rpool"> <vdev name="rpool_vdev" redundancy="none"/> </zpool> </logical> </target></pre> <p>js2ai에 의한 변환:</p> <p>root_device가 지정된 경우 js2ai는 rootdisk를 지정된 장치로 설정합니다.</p> |

표 2-2 JumpStart 프로파일 파일 키워드와 AI 매니페스트 지시어 비교 (계속)

| JumpStart 프로파일 파일 키워드 | AI 매니페스트 지시어 |
|-----------------------|---|
| system_type | AI 매니페스트는 시스템 유형을 구별하지 않습니다. js2ai에 의한 변환: standalone 값만 지원됩니다. |
| usedisk | js2ai에 의한 변환: js2ai 도구는 지정된 장치를 사용하여 이후 any 또는 rootdisk 지정을 분석할 수 있습니다. ZFS 루트 풀이 미러링되지 않은 경우, 이 목적으로 사용되지 않는 지정된 장치가 js2ai에 의해 ZFS 루트 풀에 추가됩니다. |

js2ai를 사용하여 JumpStart 규칙 및 프로파일을 AI 조건 및 매니페스트로 변환

js2ai 유틸리티를 -r 옵션과 함께 사용하면 JumpStart 규칙과 이와 연관된 프로파일이 AI 조건 및 매니페스트로 변환됩니다. 처음에는 -S 옵션을 사용하여 검증을 건너뛰십시오.

```
/usr/sbin/js2ai -rS [-d sysidcfg_dir] [-D destination_dir]
```

이 명령은 rules 파일 및 rules 파일에 참조된 프로파일에 대해 변환 작업을 수행합니다. rules 파일에 참조된 각 프로파일은 AI 클라이언트 프로비저닝 매니페스트인 /usr/share/auto_install/manifest/default.xml에 대해 처리됩니다. 이 단계를 수행하면 JumpStart rules 파일에 지정된 프로파일마다 AI_profile_name이라는 디렉토리가 생성됩니다. AI_profile_name 디렉토리에는 이 프로파일을 참조한 규칙에 해당하는 criteria-rule_number.xml 형식의 AI 조건 파일이 포함되어 있습니다. AI_profile_name 디렉토리에는 profile_name 프로파일 파일에 해당하는 profile_name.arch.xml 형식의 AI 매니페스트 파일도 포함되어 있습니다.

변환이 성공적으로 완료되었다는 메시지가 표시되지 않는 경우 오류 보고서 및 js2ai.log 파일을 확인하십시오. 오류 보고서 및 로그 파일은 경고, 처리 오류, 지원되지 않는 항목, 변환 오류 및 검증 오류를 보고합니다. 오류 보고서는 stdout에 대한 테이블 출력으로, rules 및 프로파일 파일을 변환하는 중에 발생한 각 유형의 오류 수를 표시합니다. 로그 파일에는 해당 문제가 설명되어 있습니다.

1. 프로세스 오류를 수정합니다.
2. 지원되지 않는 항목으로 나열된 rules 및 프로파일 파일에서 원하는 라인을 제거합니다.
3. 변환 오류를 확인하고 가능한 경우 오류를 수정합니다. 또는 오류가 발생한 라인을 제거합니다.
4. 경고 메시지를 확인하고 수정이 필요한지 확인합니다.

변환이 성공적으로 완료되었다는 메시지가 표시되면 -S 옵션 없이 js2ai 명령을 실행하여 출력 AI 매니페스트를 검증합니다. 검증 오류는 AI 매니페스트 파일에서 수정해야 합니다.

해당 AI DTD에 대해 특정 출력 AI 매니페스트를 검증하려면 js2ai 명령을 -V 옵션과 함께 실행하십시오.

```
/usr/sbin/js2ai -V manifest_file
```

AI 매니페스트를 AI 설치 서비스에 추가하면 해당 매니페스트도 검증됩니다.

js2ai 유틸리티를 사용하여 rules 파일 대신 하나의 JumpStart 프로파일 파일과 해당 rules 파일과 연관된 모든 프로파일을 변환하려면 -r 옵션 대신 -p 옵션을 사용하십시오.

```
/usr/sbin/js2ai -p JS_profile_name [-d sysidcfg_dir] [-D destination_dir]
```

js2ai가 JumpStart 장치 지정을 변환하는 방식

이 단원에서는 js2ai가 JumpStart 프로파일 지정 사항에서 AI 매니페스트에 대한 몇 가지 대상 요소를 결정하는 방식에 대해 설명합니다.

시스템의 루트 디스크 결정 방식

js2ai는 프로파일 변환 프로세스 중 프로파일이 참조하는 클라이언트 시스템에 액세스할 수 없으므로 js2ai는 JumpStart와 가능한 많이 일치하는 프로세스를 사용하여 변환 중 루트 디스크를 결정하려고 시도합니다.

js2ai 도구는 다음 단계를 수행하여 루트 디스크에 사용할 장치를 결정합니다.

1. 프로파일에 root_device 키워드가 지정된 경우, js2ai는 rootdisk를 슬라이스가 있는 장치로 설정합니다.
2. rootdisk가 설정되어 있지 않고 프로파일에 boot_device 키워드가 지정된 경우, js2ai는 rootdisk를 부트 장치로 설정합니다.
3. rootdisk는 설정되어 있지 않고 partitioning default는 지정되었으며 solaris fdisk 항목이 발견된 경우, js2ai는 rootdisk를 지정된 disk_name으로 설정합니다.
4. rootdisk는 설정되어 있지 않고 프로파일에 filesys cwtxdysz size / 항목이 지정된 경우, js2ai는 rootdisk를 해당 항목에 지정된 cwtxdysz 디스크로 설정합니다.
5. rootdisk는 설정되어 있지 않고 프로파일에 usedisk disk_name 항목이 지정된 경우, js2ai는 rootdisk를 해당 항목에 지정된 disk_name 디스크로 설정합니다.
6. rootdisk가 설정되어 있지 않고 size가 0이 아니거나 delete 및 disk_name이 all이 아니도록 프로파일에 지정된 경우, rootdisk는 이 disk_name으로 설정됩니다.

```
fdisk disk_name solaris size
```

7. rootdisk가 설정되어 있지 않은 경우, 장치가 rootdisk로 지정된 모든 어커런스에서 변환 오류가 발생합니다.

any 장치가 변환되는 방식

js2ai 도구는 다음 단계를 수행하여 any 키워드가 지정된 경우에 사용할 장치를 결정합니다.

1. any 장치가 지정되어 있고 키워드 작업(미러링되지 않은 풀 또는 / 마운트 지점을 포함하는 filesys)이 지정된 경우, rootdisk가 설정되어 있으면 any 장치는 rootdisk로 설정됩니다.
2. any 장치가 변환되지 않았으며 프로파일에 usedisk 문이 있는 경우, any 장치는 usedisk 문에 지정된 장치로 설정됩니다.
3. any 장치가 변환되지 않았으며 any 장치가 지정된 작업으로 인해 ZFS 루트 풀이 생성되는 경우, AI가 장치를 선택합니다. 미러링된 풀이 지정된 경우에는 해당되지 않습니다.

ZFS 루트 풀 결정 방식

js2ai 도구는 다음 단계를 수행하여 ZFS 루트 풀에 사용할 장치를 결정합니다. ZFS 루트 풀이 결정된 후 이후에 발생하는 정의가 이미 결정된 ZFS 루트 풀과 충돌할 경우 해당 정의는 오류로 플래그 지정됩니다.

1. 프로파일에 pool 키워드가 지정된 경우, js2ai는 ZFS 루트 풀을 pool 키워드에 지정된 장치로 설정합니다.
2. ZFS 루트 풀이 결정되지 않았으며 프로파일에 filesys가 / 마운트 지점과 함께 지정된 경우, 지정된 장치를 사용하여 ZFS 루트 풀이 생성됩니다.
3. ZFS 루트 풀이 결정되지 않았으며 프로파일의 모든 키워드가 처리되었고 rootdisk가 설정된 경우, rootdisk 장치를 사용하여 ZFS 루트 풀이 생성됩니다.
4. ZFS 루트 풀이 결정되지 않았으며 분할 영역 유형이 default인 경우, AI가 ZFS 루트 풀에 사용할 장치를 선택합니다.
5. ZFS 루트 풀이 결정되지 않았으며 처리 중 오류가 발생하지 않은 경우, AI가 ZFS 루트 풀에 사용할 장치를 선택합니다.
6. ZFS 루트 풀이 미러링된 풀이 아니며 하나 이상의 지정된 usedisk 장치가 rootdisk 또는 장치 변환에 사용되지 않은 경우, 해당 디스크가 ZFS 루트 풀에 추가됩니다.

소프트웨어 패키지 변환

JumpStart 프로파일은 cluster 및 package 키워드를 사용하여 시스템에 소프트웨어를 설치합니다. cluster 키워드는 AI에서 지원하지 않습니다. IPS에서 통합 또는 그룹 패키지를 설치하는 구문은 다른 패키지를 설치하는 구문과 같습니다. JumpStart 프로파일에서 단순히 cluster를 package로 변경하면 js2ai는 AI 매니페스트에 올바른 패키지 설치 지정 사항을 만듭니다.

참고 - AI 매니페스트에서 패키지 이름을 확인하십시오. AI 매니페스트에 지정된 설치 패키지를 해당 AI 매니페스트에 지정된 원본 게시자가 제공하지 않는 경우 해당 클라이언트 설치가 실패합니다.

IPS 패키지 이름은 SVR4 패키지 이름과 다릅니다. 예를 들어 SVR4 패키지 SUNWpampkcs11의 이름은 IPS에서 `library/security/pam/module/pam-pkcs11`로 바뀌었습니다.

SVR4 이름을 사용하여 IPS 패키지를 설치할 수 있도록 일부 SVR4 패키지 이름은 IPS에 있을 수 있습니다. 예를 들어 AI 매니페스트에 SUNWpampkcs11 패키지 설치가 지정된 경우, `library/security/pam/module/pam-pkcs11` 패키지가 자동으로 설치됩니다. 이 경우 패키지 이름이 바뀌었습니다.

일부 SVR4 패키지 이름은 IPS에 없습니다. 이 경우 패키지 이름을 변경하거나 AI 매니페스트에서 이러한 지정을 삭제해야 합니다. 예를 들어 SUNWCall 및 SUNWCuser 패키지의 경우 IPS에서 이름이 바뀌지 않았습니다. AI 매니페스트에 이러한 패키지가 지정된 경우 설치가 실패합니다.

js2ai 유틸리티는 `/usr/share/auto_install/manifest/default.xml` AI 매니페스트를 기반으로 JumpStart 프로파일 파일의 지정 사항을 포함하는 새 AI 매니페스트를 작성합니다. 이 기본 AI 매니페스트에는 기본 운영 체제를 설치하는 `entire` 및 `solaris-large-server`라는 두 개의 패키지 설치가 지정되어 있습니다. 사용자는 이 두 패키지 이외에 추가 도구 및 응용 프로그램 설치를 지정하기만 하면 됩니다.

Oracle Solaris 11 시스템에서 `pkg list` 명령을 사용하면 특정 패키지 이름을 AI 매니페스트에 사용할 수 있는지 여부를 확인할 수 있습니다. AI 매니페스트에 지정된 IPS 패키지 저장소 원본의 패키지를 나열하려면 `-g` 옵션을 사용해야 합니다. 이 예에서 AI 매니페스트에는 `http://pkg.oracle.com/solaris11/release`가 저장소 원본으로 지정되어 있습니다.

```
$ pkg list -af -g http://pkg.oracle.com/solaris11/release SUNWCall SUNWCuser
pkg list: no packages matching 'SUNWCuser, SUNWCall' known
```

이 메시지는 해당 두 패키지를 이 AI 매니페스트에서 사용할 수 없음을 확인합니다.

```
$ pkg list -af -g http://pkg.oracle.com/solaris11/release SUNWpampkcs11
NAME (PUBLISHER)                                VERSION                                IFO
SUNWpampkcs11                                   0.6.0-0.133                           --r
```

가장 오른쪽 열에 있는 “r”은 이 패키지의 이름이 바뀌었음을 나타냅니다. AI 매니페스트에서는 이 이름을 사용할 수 있지만, `pkg info` 명령을 사용하면 패키지의 새 이름을 확인할 수 있습니다.

다음 출력의 “Renamed to” 라인을 참조하십시오. SUNWpampkcs11 패키지의 이름은 `library/security/pam/module/pam-pkcs11`로 바뀌었습니다. 이후 Oracle Solaris 업데이트와 보다 잘 호환되도록 AI 매니페스트에 `library/security/pam/module/pam-pkcs11`을 지정할 수 있습니다.

```

$ 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: 0.173.0.0.0.487
  Packaging Date: Sun Aug 28 00:16:46 2011
  Size: 1.92 MB
  FMRI: pkg://solaris/library/security/pam/module/pam-pkcs11@...
    
```

```

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

가장 오른쪽 열에 있는 “o”는 이 패키지가 더 이상 사용되지 않음을 나타냅니다. 이 패키지는 이름은 AI 매니페스트에서 사용할 수 없습니다. pkg list 명령을 와일드카드 또는 pkg search 명령과 함께 사용하면 패키지를 사용 가능한 다른 이름으로 사용할 수 있는지 여부를 확인할 수 있습니다.

```

$ 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 매니페스트에서 SUNWmysql을 database/mysql-51로 대체할 수도 있습니다.

js2ai를 사용한 규칙 및 프로파일 변환 예

이 단원에서는 하나의 js2ai 명령을 사용하여 JumpStart rules 파일과 해당 rules 파일에서 참조하는 모든 프로파일 파일을 변환하는 방법을 보여줍니다. 기본적으로 변환된 각 프로파일은 `AI_profile-filename`이라는 디렉토리에 대한 출력입니다. 해당 프로파일을 선택하는 규칙은 AI 조건 파일과 동일한 디렉토리에 대한 출력입니다.

샘플 JumpStart 규칙 파일 및 프로파일 파일

이 예에서는 다음 rules 파일이 사용됩니다.

```

# 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 파일의 콘텐츠는 다음과 같습니다.

```

```

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

```

mirrorfilesys.profile 파일의 콘텐츠는 다음과 같습니다.

```

install_type initial_install
partitioning default

```

```
filesys mirror c6t0d0s0 c6t1d0s0 60048
cluster SUNWCuser
```

mirrorpool.profile 파일의 콘텐츠는 다음과 같습니다.

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

rootdisk.profile 파일의 콘텐츠는 다음과 같습니다.

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

규칙 파일 옵션과 함께 js2ai 사용

이 rules 파일을 처리하려면 다음 명령을 사용하십시오. 검증이 수행되지 않았으므로 오류 보고서에서 검증 오류는 하이픈으로 표시됩니다. 검증은 -s 옵션으로 생략할 수 있습니다.

```
# js2ai -rS
```

| Name | Warnings | Process Errors | Unsupported Items | Conversion Errors | Validation Errors |
|-----------------------|----------|----------------|-------------------|-------------------|-------------------|
| rules | 0 | 0 | 1 | 0 | - |
| fdisk.profile | 0 | 0 | 2 | 0 | - |
| mirrorfilesys.profile | 0 | 0 | 2 | 0 | - |
| mirrorpool.profile | 0 | 0 | 1 | 1 | - |
| rootdisk.profile | 0 | 0 | 1 | 2 | - |

Conversion completed. One or more failures 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
mirrorfilesys.profile:line 3:UNSUPPORTED: unsupported mount point of 'unnamed' specified,
mount points other than '/' and 'swap' are not supported
mirrorfilesys.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
```

출력 내용은 AI *profile-filename*에 저장됩니다. JumpStart 규칙에서 생성된 AI 조건 파일의 경우 rules 파일에 있는 규칙의 위치에 대해 이름이 지정됩니다. AI 매니페스트의 이름은 *profile-filename.arch.xml*로 지정됩니다. 여기서 *arch*는 generic, x86 또는 sparc입니다.

```
# ls AI *
AI_fdisk.profile:
fdisk.profile.x86.xml

AI_mirrorfilesys.profile:
criteria-2.xml      mirrorfilesys.profile.generic.xml

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

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

해당 AI 조건 파일

로그 파일에는 JumpStart 규칙 키워드 `hostname`이 AI 매니페스트 선택을 위한 AI 조건 키워드로 지원되지 않는다고 보고됩니다. `hostname` 키워드는 `fdisk.profile` 프로파일을 사용할 클라이언트를 지정하는 데 사용됩니다. `hostname`은 AI 매니페스트 선택 시 지원되는 키워드가 아니므로, 새 `AI_fdisk.profile` 디렉토리에는 AI 조건 파일이 포함되어 있지 않습니다.

이 문제는 `rules` 파일에서 `hostname`을 `hostaddress`로 변경하면 수정할 수 있습니다.

MAC 주소나 IP 주소를 지정하여 `hostname` 시스템을 식별하는 AI 조건 파일을 만들어 이 문제를 수정할 수도 있습니다. 예를 들어 `0:14:4F:20:53:97`이 `sample_host`의 MAC 주소인 경우 다음 조건 파일은 JumpStart 규칙 `hostname sample_host`와 동일합니다.

```
<?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>
```

시스템의 MAC 주소를 찾으려면 [Oracle Solaris 관리: 네트워크 인터페이스 및 네트워크 가상화](#) 및 `dladm(1M)` 매뉴얼 페이지에 설명된 것과 같이, `dladm` 명령을 사용하십시오.

JumpStart 규칙 `hostaddress 10.6.68.127`의 경우, `js2ai` 유틸리티가 JumpStart `hostaddress` 키워드를 AI `ipv4` 키워드로 대체하여 AI 조건 파일 `AI_mirrorfilesys.profile/criteria-2.xml`을 자동으로 만들었습니다.

```
<?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 규칙 `network 924.222.43.0`의 경우, `js2ai` 유틸리티가 제공된 네트워크 주소를 기준으로 하는 IP 주소 범위를 지정하여 AI 조건 파일 `AI_rootdisk.profile/criteria-3.xml`을 자동으로 만들었습니다.

```
<?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>
```

JumpStart 규칙 arch i386의 경우, js2ai 유틸리티가 JumpStart arch 키워드를 AI cpu 키워드로 대체하여 AI 조건 파일 AI_mirrorpool.profile/criteria-4.xml을 자동으로 만들었습니다.

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

해당 AI 매니페스트 파일

js2ai 유틸리티는 대개 오류가 보고되더라도 JumpStart 프로파일마다 AI 매니페스트를 만듭니다. 이 단원에서는 출력 AI 매니페스트가 보다 완전하도록 몇 가지 일반 오류를 해결하는 방법에 대해 설명합니다.

fdisk.profile 오류 수정

js2ai 유틸리티가 fdisk.profile JumpStart 프로파일에 대해 표시하는 오류는 다음과 같습니다.

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

이 두 라인은 무시되므로 출력 AI 매니페스트에 영향을 주지 않습니다. 변환 시 오류 메시지가 표시되지 않도록 하려는 경우 이 두 라인을 삭제할 수 있습니다. 이 경우 fdisk.profile 파일의 콘텐츠는 다음과 같습니다.

```
install_type initial_install
root device clt0d0s0
usedisk clt0d0
fdisk rootdisk solaris all
partitioning explicit
fileys rootdisk.s1 5000 swap
fileys rootdisk.s0 10000 /
```

이 프로파일만 처리하려면 js2ai 유틸리티의 -p 옵션을 사용하십시오.

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

출력 AI 매니페스트 AI_fdisk.profile/fdisk.profile.x86.xml에서 target 스탠자의 콘텐츠는 다음과 같습니다.

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

이 프로파일에는 패키지 지정 사항이 포함되어 있지 않으므로 software 스탠자는 /usr/share/auto_install/manifest/default.xml에서와 같습니다.

mirrorfileys.profile 오류 수정

js2ai 유틸리티가 mirrorfileys.profile JumpStart 프로파일에 대해 표시하는 오류는 다음과 같습니다.

```
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
```

JumpStart 프로파일 파일에서 fileys mirror 지정에서 *file_system* 매개변수는 생략할 수 없습니다. js2ai 변환에서 *file_system* 매개변수는 생략할 수 없으며 / 또는 swap 값 중 하나를 가져야 합니다.

fileys 지정의 끝에 "/"를 추가하고 cluster 라인을 삭제하도록 mirrorfileys.profile 파일을 편집하십시오. 이 경우 mirrorfileys.profile 파일의 콘텐츠는 다음과 같습니다.

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

AI_mirrorfileys.profile/criteria-2.xml 파일을 다른 위치에 저장하십시오. 그런 다음 -js2ai 유틸리티의 p 옵션을 사용하여 이 프로파일만 처리하십시오.

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

AI_mirrorfilesys.profile 디렉토리에 포함된 두 개의 출력 AI 매니페스트는 mirrorfilesys.profile.sparc.xml 및 mirrorfilesys.profile.x86.xml입니다. 이 두 AI 매니페스트 간의 유일한 차이점은 mirrorfilesys.profile.x86.xml에 두 개의 partition 지정이 포함되어 있다는 것입니다. mirrorfilesys.profile.x86.xml 매니페스트에서 target 스탠자의 콘텐츠는 다음과 같습니다.

```
<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 nodump="true" noswap="false">
    <zpool is_root="true" name="rpool">
      <vdev name="rpool_vdev" redundancy="mirror"/>
    </zpool>
  </logical>
</target>
```

이 프로파일에 패키지 지정 사항이 포함되어 있지 않으므로 software 스탠자는 /usr/share/auto_install/manifest/default.xml에서와 같습니다.

mirrorpool.profile 오류 수정

js2ai 유틸리티가 mirrorpool.profile JumpStart 프로파일에 대해 표시하는 오류는 다음과 같습니다.

```
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
```

다음 두 방법 중 하나를 사용하여 이 오류를 수정하십시오. 두 방법 모두 AI 매니페스트 출력은 동일합니다. 26 페이지 “any 장치가 변환되는 방식”도 참조하십시오.

- 두 any를 물리적 장치 이름으로 대체하여 mirrorpool.profile 프로파일을 편집합니다. cluster 라인도 삭제합니다. 이 경우 mirrorpool.profile 파일의 콘텐츠는 다음과 같습니다.

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

- pool 지정의 *vdevlist* 목록에서 물리적 장치 이름 대신 *any*를 지정한 경우, *any* 매개변수를 포함하는 지정 사항 앞에 장치 정보를 제공해야 합니다. 풀 지정 앞에 *usedisk* 지정을 추가하도록 *mirrorpool.profile* 프로파일을 편집합니다. *cluster* 라인도 삭제합니다. 이 경우 *mirrorpool.profile* 파일의 콘텐츠는 다음과 같습니다.

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

AI_mirrorpool.profile/criteria-4.xml 파일을 다른 위치에 저장하십시오. 그런 다음 *-js2ai* 유틸리티의 *p* 옵션을 사용하여 이 프로파일만 처리하십시오.

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

AI_mirrorpool.profile 디렉토리에 포함된 두 개의 출력 AI 매니페스트는 *mirrorpool.profile.sparc.xml* 및 *mirrorpool.profile.x86.xml*입니다. 이 두 AI 매니페스트 간의 유일한 차이점은 *mirrorpool.profile.x86.xml*에 두 개의 *partition* 지정이 포함되어 있다는 것입니다. *mirrorpool.profile.x86.xml* 매니페스트에서 *target* 스탠자의 콘텐츠는 다음과 같습니다.

```
<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="c6t0d0" 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 nodump="false" noswap="false">
    <zpool is_root="true" name="newpool">
      <vdev name="rpool_vdev" redundancy="mirror"/>
    </zpool>
  </logical>
</target>
```

이 프로파일에는 패키지 지정 사항이 포함되어 있지 않으므로 *software* 스탠자는 */usr/share/auto_install/manifest/default.xml*에서와 같습니다.

rootdisk.profile 오류 수정

js2ai 유틸리티가 *rootdisk.profile* JumpStart 프로파일에 대해 표시하는 오류는 다음과 같습니다.

```
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

다음 두 방법 중 하나를 사용하여 이 오류를 수정하십시오. 두 방법 모두 AI 매니페스트 출력은 동일합니다.

- rootdisk.를 c0t0d0으로 대체하여 rootdisk.profile 프로파일을 편집합니다. cluster 라인도 삭제합니다. 이 경우 rootdisk.profile 파일의 콘텐츠는 다음과 같습니다.

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

- root_device 지정을 추가하고 cluster 라인을 삭제하도록 rootdisk.profile 파일을 편집합니다. 이 경우 rootdisk.profile 파일의 콘텐츠는 다음과 같습니다.

```
install_type initial_install
partitioning explicit
root_device c0t0d0s0
fileys rootdisk.s0 15000 /
fileys rootdisk.s1 1000 swap
```

AI_rootdisk.profile/criteria-3.xml 파일을 다른 위치에 저장하십시오. 그런 다음 -js2ai 유틸리티의 p 옵션을 사용하여 이 프로파일만 처리하십시오.

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

AI_rootdisk.profile 디렉토리에 포함된 두 개의 출력 AI 매니페스트는 rootdisk.profile.sparc.xml 및 rootdisk.profile.x86.xml입니다. 이 두 AI 매니페스트 간의 유일한 차이점은 rootdisk.profile.x86.xml에 partition 지정이 포함되어 있다는 것입니다. rootdisk.profile.x86.xml 매니페스트에서 target 스탠자의 콘텐츠는 다음과 같습니다.

```
<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 nodump="true" noswap="false">
    <zpool is_root="true" name="rpool">
      <vdev name="rpool_vdev" redundancy="none"/>
    </zpool>
  </logical>
</target>
```

이 프로파일에는 패키지 지정 사항이 포함되어 있지 않으므로 software 스탠자는 /usr/share/auto_install/manifest/default.xml에서와 같습니다.

추가 프로파일 변환 정보 표시

규칙 또는 프로파일 변환이나 AI 매니페스트 검증에 대한 추가 정보를 보려면 js2ai 명령에 -v 옵션을 지정하십시오. -v 옵션을 지정하면 처리 단계가 표시되고 rules 파일 또는 프로파일을 보고서에서 생략하는 대신 오류 보고서에 0이 표시됩니다.

```
# 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

출력 AI 매니페스트 검증

생성되는 AI 매니페스트를 검증하려면 -v 옵션을 사용하십시오.

```
# 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

구성 파일 변환

이 장에서는 `sysidcfg` 파일을 AI 시스템 구성 프로파일로 변환하는 방법을 보여줍니다. 대부분의 변환은 `js2ai` 유틸리티를 사용하여 수행할 수 있습니다. [js2ai\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

AI 시스템 구성 프로파일은 시스템 구성을 지정하는 SMF XML 프로파일 파일입니다. AI 시스템 구성 프로파일에 대한 자세한 내용은 [Oracle Solaris 11.1 시스템의 11 장](#), “클라이언트 시스템 구성”을 참조하십시오.

sysidcfg 파일 키워드와 시스템 구성 프로파일 지시어 비교

다음 표는 `sysidcfg` 파일 키워드를 AI 시스템 구성 프로파일 지정 사항 예와 비교합니다. 서로 다른 SMF 등록 정보 값 이름이 서로 다른 경우에 적용될 수 있습니다. [42 페이지](#) “`js2ai`를 사용한 `sysidcfg` 변환 예” 및 [Oracle Solaris 11.1 시스템의 “예제 시스템 구성 프로파일”](#)의 예를 참조하십시오.

표 3-1 `sysidcfg` 파일 키워드와 구성 프로파일 지시어 비교

| <code>sysidcfg</code> 파일 키워드 | 시스템 구성 프로파일 지시어 |
|------------------------------|--|
| keyboard | <p><code>js2ai</code> 도구가 변환을 수행하지 않습니다. <code>sysidcfg</code> 파일에 지정된 키보드를 Oracle Solaris 11에서 지원하는지 확인합니다.</p> <p>서비스 이름: <code>system/keymap</code> 등록 정보 그룹 이름: <code>keymap</code> 등록 정보 값 이름: <code>layout</code></p> |
| name_service | <p>AI는 DNS, NIS 및 LDAP를 지원합니다. <code>js2ai</code> 도구는 NONE, DNS, NIS 및 LDAP 값을 지원합니다. NIS+ 이름 서비스는 NIS로 변환됩니다. 이름 서비스가 구성된 경우 <code>DefaultFixed</code>로 네트워크를 구성해야 합니다.</p> <p>서비스 이름: <code>system/name-service/switch</code> 등록 정보 그룹 이름: <code>config</code> 등록 정보 값 이름: <code>netgroup</code></p> |

표 3-1 sysidcfg 파일 키워드와 시스템 구성 프로파일 지시어 비교 (계속)

| sysidcfg 파일 키워드 | 시스템 구성 프로파일 지시어 |
|-------------------|--|
| network_interface | <p>AI는 시스템 설치의 일부로 단일 인터페이스만 구성할 수 있도록 지원합니다. 이 제한으로 인해 js2ai 도구는 PRIMARY 레이블의 인터페이스 또는 sysidcfg 파일에서 발견된 첫번째 인터페이스만 처리합니다. js2ai 도구는 name_service가 지정된 경우 네트워크를 DefaultFixed로 설정합니다. 제대로 구성된 DefaultFixed 네트워크는 호스트 이름, IP 주소, 넷마스크 및 게이트웨이를 제공해야 합니다. 이름 서비스가 지정되지 않은 경우 자동 네트워크 구성만 지원됩니다.</p> <p>서비스 이름: network/install 등록 정보 그룹 이름: install_ipv4_interface 등록 정보 값 이름: static_address 등록 정보 그룹 이름: install_ipv6_interface</p> |
| nfs4_domain | 지원되지 않습니다. |
| root_password | <p>js2ai 도구가 변환을 수행하지 않습니다.</p> <p>서비스 이름: system/config-user 등록 정보 그룹 이름: root_account 등록 정보 값 이름: password</p> |
| security_policy | js2ai 도구는 NONE 값을 지원합니다. |
| service_profile | js2ai 도구는 limited_net 값을 지원합니다. |
| system_locale | <p>js2ai 도구가 변환을 수행하지 않습니다. sysidcfg 파일에 지정된 로케일을 Oracle Solaris 11에서 지원하는지 확인합니다.</p> <p>서비스 이름: system/environment 등록 정보 그룹 이름: environment 등록 정보 값 이름: LC_ALL</p> |
| terminal | <p>js2ai 도구가 변환을 수행하지 않습니다. sysidcfg 파일에 지정된 터미널 유형을 Oracle Solaris 11에서 지원하는지 확인합니다.</p> <p>서비스 이름: system/console-login 등록 정보 그룹 이름: ttymon 등록 정보 값 이름: terminal_type</p> |
| timeserver | js2ai 도구는 localhost 값을 지원합니다. |
| timezone | <p>js2ai 도구가 변환을 수행하지 않습니다.</p> <p>서비스 이름: system/timezone 등록 정보 그룹 이름: timezone 등록 정보 값 이름: localtime</p> |

js2ai를 사용하여 sysidcfg 파일을 시스템 구성 프로파일로 변환

js2ai 유틸리티를 `-s` 옵션과 함께 사용하면 이 JumpStart 구성과 연관된 `sysidcfg` 파일이 시스템 구성 프로파일 파일로 변환됩니다. 처음에는 `-S` 옵션을 사용하여 검증을 건너뛰십시오.

```
/usr/sbin/js2ai -sS [-d sysidcfg_dir] [-D destination_dir]
```

처리된 각 `sysidcfg` 파일에 대해 js2ai가 `sc_profile.xml`이라는 AI 시스템 구성 프로파일 파일을 js2ai 명령이 호출된 디렉토리에 만듭니다. `sc_profile.xml` 파일에 대해 서로 다른 디렉토리를 지정하려면 `-D` 옵션을 사용하십시오.

변환이 성공적으로 완료되었다는 메시지가 표시되지 않는 경우 오류 보고서 및 `js2ai.log` 파일을 확인하십시오. 오류 보고서 및 로그 파일은 경고, 처리 오류, 지원되지 않는 항목, 변환 오류 및 검증 오류를 보고합니다. 오류 보고서는 `stdout`에 대한 테이블 출력으로, `sysidcfg` 파일을 변환하는 중에 발생한 각 유형의 오류 수를 표시합니다. 로그 파일에는 해당 문제가 설명되어 있습니다.

1. 프로세스 오류를 수정합니다.
2. 지원되지 않는 항목으로 나열된 `sysidcfg` 파일에서 원하는 라인을 제거합니다.
3. 변환 오류를 확인하고 가능한 경우 오류를 수정합니다. 또는 오류가 발생한 라인을 제거합니다.
4. 경고 메시지를 확인하고 수정이 필요한지 확인합니다.

변환이 성공적으로 완료되었다는 메시지가 표시되면 `S` 옵션 없이 `-js2ai` 명령을 실행하여 출력 `sc_profile.xml` 파일을 검증합니다. 검증 오류는 `sc_profile.xml` 파일에서 수정해야 합니다.

특정 출력 시스템 구성 프로파일을 검증하려면 js2ai 명령을 `-V` 옵션과 함께 실행하십시오.

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

시스템 구성 프로파일을 AI 설치 서비스에 추가하거나 AI 설치 서버에서 다음 명령을 실행하는 경우에도 시스템 구성 프로파일이 검증됩니다.

```
# installadm validate -n install_service_name sc_profile.xml
```

`installadm validate` 명령에 대한 자세한 내용은 **Oracle Solaris 11.1 시스템의 “시스템 구성 프로파일 검증”**을 참조하십시오.

js2ai를 사용한 sysidcfg 변환 예

이 절에서는 js2ai를 사용하여 sysidcfg 파일을 AI 시스템 구성 프로파일 파일로 변환하는 방법을 보여줍니다. 처리된 각 sysidcfg 파일에 대해 js2ai가 sc_profile.xml이라는 AI 시스템 구성 프로파일 파일을 js2ai 명령이 호출된 디렉토리에 만듭니다. sc_profile.xml 파일에 대해 서로 다른 디렉토리를 지정하려면 -D 옵션을 사용하십시오.

샘플 sysidcfg 파일

이 예에서는 다음 sysidcfg 파일이 사용됩니다.

```
timezone=US/Pacific
terminal=xterms
timeserver=localhost
network_interface=primary {
    hostname=host1
    ip_address=10.80.127.35
    netmask=255.255.255.224
    protocol_ipv6=no
    default_route=10.80.127.33}
root_password=rJmv5LUXM10cU
security_policy=none
```

sysidcfg 파일 옵션과 함께 js2ai 사용

이 sysidcfg 파일을 처리하려면 다음 명령을 사용하십시오. 검증이 수행되지 않았으므로 오류 보고서에서 검증 오류는 하이픈으로 표시됩니다. 검증은 -s 옵션으로 생략할 수 있습니다.

```
# js2ai -sS
```

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

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

```
# cat js2ai.log
```

```
sysidcfg:line 4:CONVERSION: when the PRIMARY interface is specified, by default the system will be configured for both IPv4 and IPv6 via automatic network configuration. The options specified will be ignored. If you wish to configure the interface with the specified options replace PRIMARY with the name of the interface that should be configured.
```

동등한 시스템 구성 프로파일 파일

다음과 같이 수정된 sysidcfg 파일은 js2ai.log 파일에 보고된 오류를 해결합니다. PRIMARY 인터페이스 지정은 인터페이스 이름 e1000g로 대체됩니다.

```
timezone=US/Pacific
terminal=xterms
timeserver=localhost
```

```
network_interface=e1000g {
    hostname=host1
    ip_address=10.80.127.35
    netmask=255.255.255.224
    protocol_ipv6=no
    default_route=10.80.127.33}
root_password=rJmv5LUXM10cU
security_policy=none
```

```
# js2ai -sS
Successfully completed conversion
```

이번에는 오류 보고서가 출력되지 않으므로 js2ai.log 파일이 비어 있습니다.

다음 sc_profile.xml 파일이 생성되었습니다.

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE service_bundle
  SYSTEM '/usr/share/lib/xml/dtd/service_bundle.dtd.1'>
<service_bundle name="system configuration" type="profile">
  <service name="system/timezone" type="service" version="1">
    <instance enabled="true" name="default">
      <property_group name="timezone" type="application">
        <propval name="localtime" type="astring" value="US/Pacific"/>
      </property_group>
    </instance>
  </service>
  <service name="system/console-login" type="service" version="1">
    <instance enabled="true" name="default">
      <property_group name="ttymon" type="application">
        <propval name="terminal_type" type="astring" value="xterms"/>
      </property_group>
    </instance>
  </service>
  <service name="system/config-user" type="service" version="1">
    <instance enabled="true" name="default">
      <property_group name="root_account" type="application">
        <propval name="password" type="astring" value="rJmv5LUXM10cU"/>
      </property_group>
    </instance>
  </service>
  <service name="system/identity" type="service" version="1">
    <instance enabled="true" name="node">
      <property_group name="config" type="application">
        <propval name="nodename" type="astring" value="host1"/>
      </property_group>
    </instance>
  </service>
  <service name="network/install" type="service" version="1">
    <instance enabled="true" name="default">
      <property_group name="install_ipv4_interface" type="application">
        <propval name="name" type="astring" value="e1000g/v4"/>
        <propval name="address_type" type="astring" value="static"/>
        <propval name="static_address" type="net_address_v4" value="10.80.127.35/27"/>
        <propval name="default_route" type="net_address_v4" value="10.80.127.33"/>
      </property_group>
    </instance>
```

```

</service>
<service name="network/physical" type="service" version="1">
  <instance enabled="true" name="default">
    <property_group name="netcfg" type="application">
      <propval name="active_ncp" type="astring" value="DefaultFixed"/>
    </property_group>
  </instance>
</service>
</service_bundle>

```

추가 구성 변환 정보 표시

sysidcfg 파일 변환 또는 시스템 구성 프로파일 검증에 대한 추가 정보를 보려면 js2ai 명령에 -v 옵션을 지정하십시오. -v 옵션을 지정하면 처리 단계가 표시되고 오류가 발생하지 않을 경우 오류 보고서를 생략하는 대신 오류 보고서에 0이 표시됩니다.

```

# 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 | 0 | 0 | 0 | 0 | 0 |

Successfully completed conversion

출력 시스템 구성 프로파일 검증

생성되는 시스템 구성 프로파일을 검증하려면 -v 옵션을 사용하십시오.

```

# 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

JumpStart를 사용하여 Oracle Solaris 11 서버에 Oracle Solaris 10 설치

JumpStart를 사용하여 네트워크 SPARC 및 x86 플랫폼에 Oracle Solaris 10 운영 체제를 설치하는 데 익숙한 사용자라면 JumpStart로는 Oracle Solaris 11 OS가 아닌 Oracle Solaris 10 OS만 설치할 수 있다는 점을 알고 있을 것입니다. 그러나 JumpStart 설치 서버는 Oracle Solaris 11 시스템에 설치할 수 있습니다.

Oracle Solaris 11 서버는 두 개의 다른 작업을 수행할 수 있습니다.

- 자동 설치 프로그램을 사용하여 Oracle Solaris 11 OS 설치를 제공합니다. 자세한 내용은 [Oracle Solaris 11.1 시스템의 제III부, “설치 서버를 사용하여 설치”](#)를 참조하십시오.
- JumpStart를 사용하여 Oracle Solaris 10 OS 설치를 제공합니다. 이 장에서는 Oracle Solaris 11 시스템에 JumpStart 설치 서버를 설정하는 방법에 대해 설명합니다. JumpStart에 대한 자세한 내용은 [Oracle Solaris 10 8/11 설치 설명서: 사용자 정의 JumpStart 및 고급 설치](#)를 참조하십시오.

Oracle Solaris 11 시스템을 Oracle Solaris 10 JumpStart 서버로 설정

다음 단계는 Oracle Solaris 11 JumpStart 설치 서버를 만들어 클라이언트 시스템에 Oracle Solaris 10 OS를 설치하는 방법에 대해 설명합니다.

1. JumpStart 설치 서버가 될 시스템에 Oracle Solaris 11 OS를 설치합니다.
2. JumpStart 설치 서버가 될 Oracle Solaris 11 시스템에 정적 IP 주소를 설정합니다.
정적 IP 주소를 설정하는 방법은 [Oracle Solaris 관리: 네트워크 인터페이스 및 네트워크 가상화의 “IP 인터페이스를 구성하는 방법”](#) 및 `ipadm(1M)` 매뉴얼 페이지의 지침을 참조하십시오.
3. `solaris` 게시자에서 `system/boot/network` 패키지를 설치합니다.

```
# 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. 최신 [Oracle Solaris 10 DVD](#) 이미지를 다운로드합니다.

이 이미지는 설치 이미지 및 JumpStart 도구의 소스입니다. JumpStart로는 Oracle Solaris 11 OS가 아닌 Oracle Solaris 10 OS만 설치할 수 있다는 점을 유념하십시오.

5. Oracle Solaris 10 DVD 이미지 및 `setup_install_server(1M)` 명령을 사용하여 JumpStart 설치 서버를 설정합니다.

다음 예에서는 SPARC용 Oracle Solaris 10 9/10 DVD 이미지를 사용합니다.

```

# /media/SOL_10_0910_SPARC/Solaris_10/Tools/setup_install_server /export/s10u9_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. DVD 이미지의 `jumpstart_sample` 파일을 JumpStart 프로파일 위치로 복사합니다.

```

# cd /media/SOL_10_0910_SPARC/Solaris_10/Misc/jumpstart_sample
# mkdir -p /export/profiles/s10profile
# cp -pr * /export/profiles/s10profile

```

7. [48 페이지 “rm_install_client 스크립트 문제”](#)에 설명된 임시해결책을 수행합니다.
8. [48 페이지 “check 스크립트 문제”](#)에 설명된 임시해결책을 수행합니다.
9. `check(1M)` 명령을 `-p s10-image` 옵션과 함께 실행하여 JumpStart 프로파일을 검증합니다.

다음 예에서는 간략하게 보이기 위해 출력의 여러 라인을 생략했습니다. `-p` 옵션은 일부 Oracle Solaris 10 업데이트 릴리스에서만 작동합니다. Oracle Solaris 10 업데이트 7 이상을 사용하고 있는지 확인합니다 (CR 6728067 참조).

```

# cd /export/profile/s10profile
./check -p /export/s10u9_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.
```

10. udp6이 온라인 상태로 사용 가능한지 확인합니다.

udp6 서비스는 tftp 패키지로 제공됩니다. tftp 패키지가 설치되었는지 확인합니다.

```
# pkg list tftp
```

tftp 패키지가 설치되지 않은 경우 다음 명령을 사용하여 설치합니다.

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

udp6 서비스가 온라인 상태로 사용 가능한지 확인합니다.

```
# 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
```

11. /tftpboot 디렉토리를 /etc/netboot 디렉토리에 링크합니다. /etc/netboot 디렉토리가 없으면 만듭니다.

```
# 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
```

JumpStart는 네트워크 부트 시 /tftpboot를 사용합니다. AI는 네트워크 부트 시 /etc/netboot를 사용합니다. 이 Oracle Solaris 11 서버가 JumpStart 설치 서버와 AI 설치 서버로 사용될 경우, /tftpboot가 /etc/netboot에 대한 심볼릭 링크여야 두 도구가 동시에 존재할 수 있습니다.

참고 - 이 서버를 AI 설치 서버에서 JumpStart 설치 서버로만 사용하려는 경우에는 이 단계가 필요하지 않습니다. 그러나 나중에 계획을 변경할 경우 netboot 실패 오류를 방지하려면 이 단계를 수행해야 합니다.

12. add_install_client(IM)를 실행하여 클라이언트를 설치합니다.

```
/export/s10u9_sparc/Solaris_10/Tools/add_install_client -e 8:0:20:fd:f2:18 \
-c line2-x4100:/export/profiles/s10profile \
-p line2-x4100:/export/profiles/s10profile line2-t1 sun4u
Adding Ethernet number for line2-t1.sfbay.sun.com to /etc/ethers
making /tftpboot
```

```
enabling network/rarp service
enabling network/rpc/bootparams service
updating /etc/bootparams
copying boot file to /tftpboot/inetboot.SUN4U.Solaris_10-1
```

알려진 문제에 대한 임시해결책

이 단원에서는 특정 문제를 방지하기 위한 지침을 제공합니다.

rm_install_client 스크립트 문제

CR 6646677: The `rm_install_client` script does not work with Solaris 10 dig based nslookup

이 문제를 해결하려면 Oracle Solaris 10 넷 이미지의 `rm_install_client` 스크립트에서 다음과 같이 변경하십시오. 예를 들어 `/export/s10u9/dvds/latest`에서 Oracle Solaris 10 넷 이미지를 설정한 경우, `/export/s10u9/dvds/latest/Solaris_10/Tools/rm_install_client`에 있는 `rm_install_client` 스크립트에서 다음과 같이 변경하십시오.

다음에서 라인 168을

```
ANS='nslookup ${K} 2>&1'
```

다음과 같이 변경하십시오.

```
ANS='nslookup ${K} | /bin/sed '/^;;/d' 2>&1'
```

다음에서 라인 273을

```
ANS='echo $ANS | sed -e 's/#.*$//''
```

다음과 같이 변경하십시오.

```
ANS='echo $ANS | sed -e 's/ #.*$//''
```

check 스크립트 문제

CR 6838095: opensolaris jumpstart server doesn't run the check script correctly

이 문제를 해결하려면 `/usr/sunos/bin/sh`를 사용하도록 JumpStart 프로파일 디렉토리 `/export/profile/s10profile`에서 `check` 스크립트를 변경하십시오. 이때 `check` 스크립트의 첫번째 라인은 다음 라인으로 대체하십시오.

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