

# **Guide du programme d'installation automatisée Oracle Solaris 11 Express**

Copyright © 2008, 2010, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. UNIX est une marque déposée concédée sous licence par X/Open Company, Ltd.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.

---

Copyright © 2008, 2010, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. UNIX est une marque déposée concédée sous licence par X/Open Company, Ltd.

# Table des matières

---

<b>Préface</b> .....	7
<b>1 Présentation du programme d'installation automatisée</b> .....	11
Qu'est-ce qu'une installation automatisée ? .....	11
Comment utiliser le programme d'installation automatisée ? .....	12
<b>2 Configuration d'un serveur d'installation IA</b> .....	15
Liste des tâches de configuration du serveur IA .....	15
Configuration requise pour le serveur d'installation .....	16
Configuration matérielle requise pour le serveur IA .....	16
Configuration logicielle requise pour le serveur IA .....	16
Configuration de l'image d'installation et du service d'installation .....	17
Recherche d'espace pour vos fichiers d'image IA .....	18
Installation des outils d'installation IA .....	18
Configuration d'une image d'initialisation IA .....	19
Création d'un service d'installation IA .....	20
Consultez les instructions d'installation par défaut .....	23
Gestion d'un serveur d'installation .....	24
Ajout ou suppression d'un service d'installation .....	25
Activation ou désactivation d'un service d'installation .....	25
Association de clients à des services d'installation .....	25
Association d'instructions d'installation spécifiques au client à des services d'installation .....	26
Affichage des informations relatives aux services d'installation .....	29
Affichage des informations sur les installations personnalisées .....	30
Administration du service SMF IA .....	31

<b>3</b>	<b>Personnalisation des installations</b> .....	33
	Mise en correspondance des instructions d'installation avec les caractéristiques de clients .....	33
	Fichier manifest IA par défaut .....	37
<b>4</b>	<b>Spécification des instructions d'installation</b> .....	41
	Balises de fichier manifest IA .....	41
	Création d'un fichier manifest IA personnalisé .....	44
	Définition d'une cible pour l'installation .....	45
	Spécifications déterministes de disque cible .....	47
	Spécifications non déterministes de disque cible .....	50
	Configuration du partitionnement sur un client x86 .....	51
	Configuration de tranches sur un disque .....	55
	Configuration du swap et du vidage sur le périphérique d'installation .....	57
	Installation de logiciels .....	59
	Spécification d'une source de packages à installer .....	59
	Spécification des packages à installer .....	61
	Identification et installation de pilotes manquants sur une cible d'installation .....	63
	Fichier manifest IA annoté .....	67
<b>5</b>	<b>Configuration du système client</b> .....	73
	Création d'un fichier manifest SC personnalisé .....	73
	Spécification de la configuration dans un fichier manifest SC .....	74
	Comptes root et utilisateur .....	74
	Type de terminal et disposition du clavier .....	76
	Adresse IP statique et DNS .....	76
	Exemples de fichiers manifest SC .....	79
	Spécification du type de terminal et de la disposition du clavier .....	79
	Spécification de la configuration réseau statique .....	81
<b>6</b>	<b>Configuration de DHCP pour le programme d'installation automatisée</b> .....	83
	DHCP Oracle Solaris pour le programme d'installation automatisée .....	83
	Configuration d'un serveur DHCP sur le serveur d'installation IA .....	83
	Configuration d'un serveur DHCP distinct .....	84
	Configuration d'un serveur DHCP existant pour le programme d'installation automatisée .....	84

ISC DHCP pour le programme d'installation automatisée .....	86
Configuration réseau de base .....	86
Configuration de DNS .....	86
Configuration du serveur d'initialisation et du fichier d'initialisation .....	87
Exemple de fichier de configuration .....	88
<b>7 Installation de systèmes clients .....</b>	<b>89</b>
Installation d'un client .....	89
Configuration système requise pour les clients .....	90
Configuration système requise pour les clients SPARC et x86 .....	90
Configuration système requise supplémentaire pour les clients SPARC .....	91
Configuration d'un client d'installation .....	92
Configuration d'un client x86 .....	92
Configuration d'un client SPARC .....	93
Suppression d'un client à partir d'un service .....	94
Installation de clients .....	94
Utilisation de Secure Shell pour contrôler à distance les installations .....	94
Installation d'un client SPARC .....	95
Installation d'un client x86 .....	97
Messages d'installation du client .....	99
<b>8 Installations automatisées initialisées à partir d'un support .....</b>	<b>101</b>
Présentation de l'installation effectuée à partir d'un support IA .....	101
Installation à partir d'un support IA .....	101
Configuration système requise pour l'installation à partir d'un support IA .....	102
Installation à partir d'un support IA .....	103
<b>A Dépannage des installations automatisées .....</b>	<b>107</b>
Échec de l'installation du client .....	107
Vérifier les journaux d'installation .....	107
Vérifier le référentiel IPS .....	107
Vérifier DNS .....	108
Vérifier les erreurs d'initialisation du client .....	109
Run Automated Installations in Debug Verbose Mode (Exécuter les installations automatisées en mode de débogage détaillé) .....	116

Boot SPARC Systems in Debug Mode (Initialiser les systèmes SPARC en mode de débogage) .....	117
Boot x86 Systems in Debug Mode (Initialiser les systèmes x86 en mode de débogage) ....	117
Boot the Install Environment Without Starting an Installation (Initialiser l'environnement d'installation sans démarrer d'installation) .....	117
Start Installation After Booting Without Initiating an Installation (Démarrer l'installation après l'initialisation sans lancer une installation) .....	118
<b>B Commandes d'administration de l'installation automatisée .....</b>	<b>119</b>
Page de manuel installadm(1M) .....	119
<b>C Migration de JumpStart vers le programme d'installation automatisée .....</b>	<b>131</b>
Comparaison de JumpStart et du programme d'installation automatisée .....	131
Conversion des règles JumpStart en critères IA .....	132
Comparaison entre les mots-clés de règles et les directives de critères .....	132
Conversion des fichiers de règles en critères .....	134
Conversion d'un profil JumpStart en fichier manifest IA .....	136
Conversion des fichiers de profil en fichiers manifest IA .....	138

# Préface

---

Le *Guide du programme d'installation automatisée Oracle Solaris 11 Express* fournit les instructions d'utilisation du programme d'installation automatisée (IA) afin d'installer le système d'exploitation Oracle Solaris sur plusieurs systèmes client sur un réseau. Vous pouvez également utiliser le support IA pour installer le SE Oracle Solaris sur une plate-forme SPARC ou x86. Dans tous les cas, l'accès à un référentiel de packages sur le réseau est nécessaire pour terminer l'installation.

## Utilisateurs de ce manuel

Ce manuel s'adresse aux administrateurs système qui veulent effectuer une installation mains-libres du système d'exploitation Oracle Solaris 11 Express sur le réseau.

## Organisation de ce document

Ce manuel contient les chapitres et annexes suivants :

- [Chapitre 1, “Présentation du programme d'installation automatisée”](#) : donne un aperçu très détaillée de la procédure d'installation à l'aide du programme d'installation automatisée.
- [Chapitre 2, “Configuration d'un serveur d'installation IA”](#) : décrit la configuration et la gestion d'un serveur d'installation et les services d'installation.
- [Chapitre 3, “Personnalisation des installations”](#) : décrit l'exécution de différentes installations sur plusieurs clients en une seule installation du programme d'installation automatisée.
- [Chapitre 4, “Spécification des instructions d'installation”](#) : décrit en détail comment spécifier la méthode souhaitée d'installation des clients.
- [Chapitre 5, “ Configuration du système client”](#) : décrit en détail comment spécifier la configuration du client après installation dans le cadre de l'installation automatisée.
- [Chapitre 6, “Configuration de DHCP pour le programme d'installation automatisée”](#) : décrit la configuration des serveurs Oracle Solaris DHCP et ISC DHCP. L'installation automatisée nécessite le protocole DHCP (Dynamic Host Configuration Protocol), avec la résolution de nom DNS (Domain Name System).
- [Chapitre 7, “Installation de systèmes clients”](#) : décrit l'association d'un client particulier avec un service d'installation spécifique et la méthode de lancement de l'installation automatisée.

- [Chapitre 8, “Installations automatisées initialisées à partir d’un support”](#) : décrit la procédure d’installation effectuée à l’aide d’un support IA.
- [Annexe A, “Dépannage des installations automatisées”](#) : fournit de l’aide en vue du diagnostic et de la résolution des problèmes.
- [Annexe B, “Commandes d’administration de l’installation automatisée”](#) fournit la page de manuel `instaladm(1M)`.
- [Annexe C, “Migration de JumpStart vers le programme d’installation automatisée”](#) fournit des informations pour faciliter la migration de Solaris Custom JumpStart vers le programme d’installation automatisée Oracle Solaris.

## Informations connexes

Outre la documentation répertoriée, reportez-vous à la documentation relative à l’administrateur système Oracle Solaris 11 Express pour plus d’informations sur le mode d’administration des systèmes Oracle Solaris.

- [Démarrage d’Oracle Solaris 11 Express](#) explique comment installer le système d’exploitation Oracle Solaris 11 Express à l’aide du CD-ROM Live ou du programme d’installation en mode texte.
- [Guide d’Oracle Solaris 11 Express Image Packaging System](#) explique comment utiliser les éditeurs IPS et les référentiels de packages et comment installer et mettre à jour le logiciel après la première installation du système. Voir aussi `pkg(1)`.
- [System Administration Guide: IP Services](#) traite de DHCP Oracle Solaris.
- [System Administration Guide: Naming and Directory Services \(DNS, NIS, and LDAP\)](#) traite de DNS Oracle Solaris.
- [Oracle Solaris ZFS Administration Guide](#) explique comment configurer et administrer les systèmes de fichiers ZFS Oracle Solaris. Voir également `zfs(1M)` et `zpool(1M)`.
- [Service Management Facility \(SMF\) sur Oracle Technology Network](#), y compris :
  - [Utilitaire de gestion des services Solaris - Guide de démarrage rapide](#)
  - [Introduction au développeur de services de l’utilitaire de gestion des services Solaris \(SMF, Service Management Facility\)](#)
  - [How to Create a Service Management Facility \(identification requise\)](#)



## Documentation, support et formation

Pour obtenir des ressources supplémentaires, reportez-vous aux sites Web suivants :

- [Documentation](http://docs.sun.com) (<http://docs.sun.com>)
- [Assistance](http://www.oracle.com/us/support/systems/index.html) (<http://www.oracle.com/us/support/systems/index.html>)
- [Formation](http://education.oracle.com) (<http://education.oracle.com>) - Cliquez sur le lien Sun dans la barre de navigation à gauche.

## Ressources logicielles Oracle

Le site [Oracle Technology Network](http://www.oracle.com/technetwork/index.html) (<http://www.oracle.com/technetwork/index.html>) propose une large gamme de ressources pour les logiciels Oracle :

- Discutez de problèmes techniques et trouvez des solutions sur les [forums de discussion](http://forums.oracle.com) (<http://forums.oracle.com>).
- Passez à la pratique grâce aux didacticiels étape par étape avec [Oracle By Example](http://www.oracle.com/technetwork/tutorials/index.html) (<http://www.oracle.com/technetwork/tutorials/index.html>).
- Téléchargez un [échantillon de code](http://www.oracle.com/technology/sample_code/index.html) ([http://www.oracle.com/technology/sample\\_code/index.html](http://www.oracle.com/technology/sample_code/index.html)).

## Conventions typographiques

Le tableau ci-dessous décrit les conventions typographiques utilisées dans ce manuel.

TABLEAU P-1 Conventions typographiques

Type de caractères	Signification	Exemple
AaBbCc123	Noms des commandes, fichiers et répertoires, ainsi que messages système.	Modifiez votre fichier <code>.login</code> .  Utilisez <code>ls -a</code> pour afficher la liste de tous les fichiers.  <code>nom_machine%</code> Vous avez reçu du courrier.
<b>AaBbCc123</b>	Ce que vous entrez, par opposition à ce qui s'affiche à l'écran.	<code>nom_machine%</code> <b>su</b>  Mot de passe :
<i>aabbcc123</i>	Paramètre fictif : à remplacer par un nom ou une valeur réel(le).	La commande permettant de supprimer un fichier est <code>rm nom_fichier</code> .

TABLEAU P-1 Conventions typographiques (Suite)

Type de caractères	Signification	Exemple
<i>AaBbCc123</i>	Titres de manuel, nouveaux termes et termes importants.	Reportez-vous au chapitre 6 du <i>Guide de l'utilisateur</i> .  Un <i>cache</i> est une copie des éléments stockés localement.  <i>N'enregistrez pas</i> le fichier.  <b>Remarque</b> : en ligne, certains éléments mis en valeur s'affichent en gras.

## Invites de shell dans les exemples de commandes

Le tableau suivant présente l'invite système UNIX par défaut et l'invite superutilisateur pour les shells faisant partie du SE Oracle Solaris. L'invite système par défaut qui s'affiche dans les exemples de commandes dépend de la version Oracle Solaris.

TABLEAU P-2 Invites de shell

Shell	Invite
Bash shell, korn shell et bourne shell	\$
Bash shell, korn shell et bourne shell pour superutilisateur	#
C shell	nom_machine%
C shell pour superutilisateur	nom_machine#

# Présentation du programme d'installation automatisée

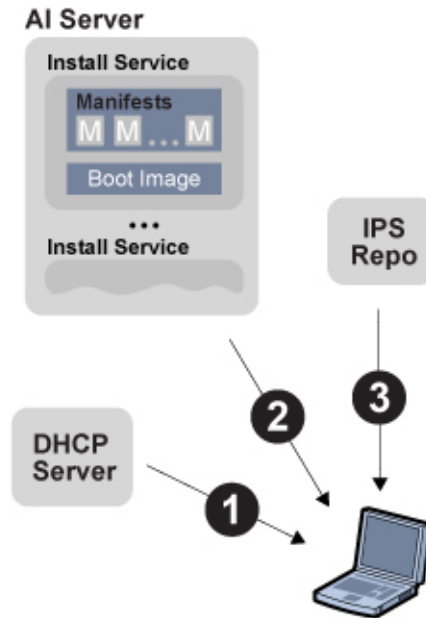
---

Utilisez le programme d'installation automatisée (IA) pour installer le système d'exploitation Oracle Solaris (SE) sur plusieurs systèmes client d'un réseau. Le programme d'installation automatisée effectue une installation "mains libres" des systèmes x86 et SPARC. Vous pouvez également utiliser le support IA pour installer le SE Oracle Solaris sur une seule plate-forme SPARC ou x86. Dans tous les cas, l'accès à un référentiel de packages sur le réseau est nécessaire pour terminer l'installation.

## Qu'est-ce qu'une installation automatisée ?

Le programme d'installation automatisée automatise l'installation du SE Oracle Solaris sur un ou plusieurs clients SPARC ou x86 d'un réseau. Les clients peuvent diverger en termes d'architecture, de disque et de capacité de mémoire, et d'autres paramètres. Les installations peuvent diverger en termes de configuration réseau, de packages installés, de capacité du disque et d'autres spécifications.

FIGURE 1-1 Installation automatisée des clients sur un réseau



Une installation automatisée d'un client sur un réseau local se compose des étapes de haut niveau suivantes :

1. Un système client s'initialise et récupère les informations IP à partir du serveur DHCP.
2. Les caractéristiques du client déterminent le service IA et les instructions d'installation utilisés pour l'installer.
3. Le SE Oracle Solaris est installé sur le client par le biais de l'extraction des packages du référentiel de packages spécifié dans les instructions d'installation du service IA.

## Comment utiliser le programme d'installation automatisée ?

Pour utiliser le programme d'installation automatisée pour installer les systèmes client sur le réseau, vous devez configurer DHCP et un service IA sur un serveur IA. Reportez-vous au [Chapitre 2, “Configuration d'un serveur d'installation IA”](#). Le programme d'installation automatisée utilise DHCP pour indiquer l'adresse IP, le masque de sous-réseau, le routeur, le serveur DNS et l'emplacement d'une image d'initialisation à l'ordinateur client à installer. Les serveurs DHCP et IA peuvent être le même ordinateur ou deux ordinateurs différents. Reportez-vous au [Chapitre 6, “Configuration de DHCP pour le programme d'installation automatisée”](#).

Les machines client que vous voulez installer doivent être en mesure d'accéder à un référentiel de packages Oracle Solaris Image Packaging System (IPS). Le référentiel de packages IPS peut se trouver sur le serveur IA ou sur un autre serveur sur le réseau local. Il peut également se trouver sur Internet.

Un service IA est associé à une image d'installation IA SPARC ou x86 et à un ou plusieurs ensembles d'instructions d'installation. L'image IA n'est pas une installation complète. Les machines client doivent accéder à un référentiel de packages IPS pour terminer leur installation. Les instructions d'installation spécifient un ou plusieurs référentiels de packages IPS dans lesquels le client récupère les packages nécessaires pour terminer l'installation. Les instructions d'installation comprennent également les noms des autres packages à installer et des informations, notamment sur le périphérique cible et la partition. Reportez-vous au [Chapitre 4, "Spécification des instructions d'installation"](#) pour de plus amples informations. Vous pouvez également spécifier des instructions de configuration post-installation du client. Reportez-vous au [Chapitre 5, "Configuration du système client"](#).

Si deux machines client sont dotées d'architectures différentes ou doivent être installées avec des versions différentes du SE Oracle Solaris, vous devez créer deux services IA et associer chacun d'eux à une image IA différente.

Si deux machines client doivent être installées avec la même version du SE Oracle Solaris mais requièrent une installation différente, créez deux ensembles d'instructions d'installation pour le service IA. Les différentes instructions d'installation peuvent spécifier différents packages à installer ou une tranche différente comme cible d'installation, par exemple.

L'installation démarre lorsque vous initialisez le client. Lorsque le client démarre, DHCP le dirige vers le serveur d'installation IA. Il accède au service d'installation correct et aux instructions d'installation correctes au sein de ce service. Le [Chapitre 7, "Installation de systèmes clients"](#) explique comment un client est associé à un service d'installation spécifique. Le [Chapitre 3, "Personnalisation des installations"](#) explique comment un client trouve les instructions d'installation appropriées à utiliser.



# Configuration d'un serveur d'installation IA

---

Pour installer des clients sur un réseau, le programme d'installation automatisée utilise un serveur d'installation distinct. Sur le serveur IA, stockez une image d'installation IA et créez un service IA. Le service IA spécifie les instructions d'installation du SE Oracle Solaris sur différents clients.

## Liste des tâches de configuration du serveur IA

La liste des tâches ci-après résume les étapes à suivre pour configurer un serveur IA.

TABLEAU 2-1 Liste des tâches

Tâche	Référence
Vérifiez si le serveur répond à la configuration matérielle minimale requise pour être utilisé en tant que serveur IA.	Reportez-vous à la section <a href="#">“Configuration matérielle requise pour le serveur IA”</a> à la page 16.
Assurez-vous que le serveur IA exécute le système d'exploitation Oracle Solaris 11 Express. Vérifiez les autres logiciels requis pour le serveur IA.	Reportez-vous à la section <a href="#">“Configuration logicielle requise pour le serveur IA”</a> à la page 16.
Configurez le serveur IA de sorte qu'il utilise une adresse IP statique.	Reportez-vous à la section <a href="#">“Configuration logicielle requise pour le serveur IA”</a> à la page 16.
Installez l'ensemble d'outils IA.	Reportez-vous à la section <a href="#">“Installation des outils d'installation IA”</a> à la page 18.
Obtenez une image d'installation IA.	Reportez-vous à la section <a href="#">“Configuration d'une image d'initialisation IA”</a> à la page 19.
Configurez un service d'installation.	Reportez-vous à la section <a href="#">“Création d'un service d'installation IA”</a> à la page 20. Vous avez besoin d'un service d'installation distinct pour chaque architecture à installer, un service d'installation n'étant associé qu'à une seule image d'installation.

TABLEAU 2-1 Liste des tâches (Suite)

Tâche	Référence
Vérifiez les spécifications d'installation par défaut.	Si ces spécifications ne sont pas adaptées à vos clients, vous pouvez personnaliser les instructions. Reportez-vous à la section <a href="#">“Consultez les instructions d'installation par défaut”</a> à la page 23.

## Configuration requise pour le serveur d'installation

Tous les systèmes qui correspondent à cette configuration requise peuvent être utilisés en tant que serveurs IA, y compris les ordinateurs portables, les ordinateurs de bureau, les machines virtuelles et les serveurs d'entreprise. Le serveur IA peut être soit une machine x86 ou SPARC. Un serveur d'installation x86 peut installer les clients SPARC et x86, et un serveur d'installation SPARC peut installer les clients SPARC et x86.

## Configuration matérielle requise pour le serveur IA

Si vous devez installer ou mettre à jour la version Oracle Solaris 11 Express sur votre serveur d'installation IA, consultez la documentation de l'installation pour connaître la configuration requise en termes de mémoire et d'espace disque.

**Mémoire** La configuration minimale requise pour un fonctionnement en tant que serveur d'installation IA est de 1 Go de mémoire.

**Espace disque** Une fois la version Oracle Solaris 11 Express installée sur votre serveur d'installation IA, vous avez besoin d'environ 0,75 Go d'espace disque supplémentaire pour chaque service d'installation IA que vous souhaitez créer. Vous avez besoin d'un service d'installation distinct pour chaque architecture client différente et chaque version différente du système d'exploitation que vous envisagez d'installer. Reportez-vous à la section [“Configuration de l'image d'installation et du service d'installation”](#) à la page 17.

## Configuration logicielle requise pour le serveur IA

**Système d'exploitation** Installez la version Oracle Solaris 11 Express sur le serveur IA. Pour ce faire, reportez-vous à la section [\*Démarrage d'Oracle Solaris 11 Express\*](#) ou au [Chapitre 8, “Installations automatisées initialisées à partir d'un support”](#).



<b>Adresse IP statique</b>	Configurez le serveur IA de sorte qu'il utilise une adresse IP statique. Reportez-vous à la section <a href="#">“How to Configure an IP Interface”</a> du <i>System Administration Guide: Network Interfaces and Network Virtualization</i> .
<b>Routeur par défaut</b>	Assurez-vous qu'une route par défaut est définie pour votre serveur IA en affichant l'état du réseau à l'aide de la commande <code>netstat(1M)</code> . Si une route par défaut n'est pas définie pour votre serveur IA, vous pouvez définir une en remplissant le fichier <code>/etc/default/router(4)</code> avec l'adresse IP d'une route statique par défaut pour le réseau de votre serveur.
<b>DHCP</b>	Configurez un serveur DHCP. Le serveur IA peut également être utilisé en tant que serveur DHCP. Vous pouvez également utiliser un serveur DHCP qui est déjà configuré dans ce réseau. Vous avez besoin de différentes configurations DHCP pour chaque architecture client. La section <a href="#">“Création d'un service d'installation comprenant la configuration du serveur DHCP Oracle Solaris”</a> à la page 22 et le Chapitre 6, <a href="#">“Configuration de DHCP pour le programme d'installation automatisée”</a> fournissent des informations sur l'installation et la configuration du serveur DHCP pour le programme d'installation automatisée.

## Configuration de l'image d'installation et du service d'installation

Effectuez les étapes ci-après pour configurer une image d'installation et un service d'installation :

- Installez l'ensemble d'outils d'installation IA.
- Téléchargez une image d'initialisation IA.
- Créez un service d'installation IA.

## Recherche d'espace pour vos fichiers d'image IA

Les fichiers ISO de l'image d'initialisation IA et les versions étendues de ces images doivent être stockés sur votre serveur IA. Prévoyez de stocker ces fichiers sur un système de fichiers ZFS distinct afin de tirer parti des fonctions ZFS, par exemple :

- Définissez des caractéristiques de systèmes de fichiers distincts, comme par exemple `compression` ou `atime`.
- Créez directement des instantanés des systèmes de fichiers spécifiés afin de pouvoir les restaurer.

Reportez-vous au [Oracle Solaris ZFS Administration Guide](#) pour obtenir plus d'informations et connaître les meilleures pratiques en matière de création de systèmes de fichiers ZFS.

## Installation des outils d'installation IA

Le package des outils d'installation IA fournit les commandes `installadm(1M)` qui permettent de créer et de gérer les services d'installation IA.

L'utilitaire `installadm` permet d'effectuer les tâches suivantes :

- Créez et activez les services d'installation.
- Configurez et mettez à jour un serveur DHCP.
- Ajoutez des instructions d'installation et de configuration des clients personnalisée.
- Définissez des critères de sorte que les clients utilisent les instructions d'installation et de configuration personnalisées.

Reportez-vous à la section “[Gestion d'un serveur d'installation](#)” à la page 24 et à l'[Annexe B](#), “[Commandes d'administration de l'installation automatisée](#)” pour plus d'informations concernant les commandes `installadm`.

Pour installer le package d'outils, votre serveur d'installation IA doit être en mesure d'accéder à un référentiel de packages Image Packaging System (IPS) Oracle Solaris. Assurez-vous que vous êtes connecté à Internet ou à un serveur de packages IPS local qui contient le package `installer/installadm`. La version du package `installer/installadm` que vous installez doit être la même que la version du SE Oracle Solaris exécutée par le serveur d'installation IA.

Assurez-vous que votre éditeur préféré est défini dans le référentiel à partir duquel vous souhaitez obtenir le package d'outils IA, ou indiquez-le lorsque vous installez celui-ci.

L'exemple suivant indique que deux éditeurs sont définis :

```
# pkg publisher
PUBLISHER                TYPE    STATUS  URI
```

```
solaris      (preferred) origin online http://pkg.oracle.com/solaris/release
example.com (non-sticky) origin online http://pkg.example.com/
```

L'exemple ci-après installe le package d'outils IA à partir de l'éditeur préféré, `solaris` :

```
# pkg install install/installadm
```

L'exemple ci-après installe le package d'outils à partir de l'éditeur `example.com`, qui n'est pas l'éditeur préféré :

```
# pkg install pkg://example.com/install/installadm
```

Reportez-vous au *Guide d'Oracle Solaris 11 Express Image Packaging System* pour plus d'informations sur l'ajout d'un éditeur et la définition d'un éditeur préféré.

## Configuration d'une image d'initialisation IA

Le programme d'installation automatisée utilise une image d'initialisation minimale pour initialiser le client. Après l'initialisation du client, l'installation du SE Oracle Solaris se poursuit selon les instructions d'installation figurant dans le service d'installation.

### ▼ Comment configurer une image d'initialisation IA

#### 1 Assumez le rôle root :

```
$ su - root
```

#### 2 Créez un système de fichiers ZFS dans le pool root pour les images d'initialisation.

Dans cet exemple, le système de fichiers ZFS est nommé `ai`.

```
# zfs create rpool/ai
```

Dans de nombreux cas, la définition de l'option `compression=on` est recommandée. Dans ce cas, la spécification de la compression peut représenter un avantage limité parce que les fichiers ISO IA sont déjà compressés. De même, la définition de `dedup=on` peut être conseillée. Reportez-vous à la section “*Introducing ZFS Properties*” du *Oracle Solaris ZFS Administration Guide* pour plus d'informations sur les propriétés du système de fichiers ZFS. Reportez-vous à la section “*The dedup Property*” du *Oracle Solaris ZFS Administration Guide* pour plus d'informations sur la déduplication.

#### 3 Téléchargez l'image d'initialisation.

Pour télécharger l'image d'initialisation IA, allez à la page Web suivante :

```
http://www.oracle.com/technetwork/server-storage/solaris11/downloads/index.html
```

Assurez-vous de télécharger l'image Automated Install et non l'image Live CD ou l'image d'installation texte. Téléchargez l'image Oracle Solaris 11 Express SPARC Automated Install pour les clients SPARC ou l'image Oracle Solaris 11 Express x86 Automated Install pour les clients x86. Pour les clients x86, veillez à télécharger le fichier `.iso` et non le fichier `.usb`. Le fichier `.usb` peut être utilisé pour lancer une installation IA par le biais de l'initialisation d'un système à partir d'un périphérique USB. Reportez-vous au [Chapitre 8, "Installations automatisées initialisées à partir d'un support"](#). Le fichier `.usb` ne convient pas à la création d'un service d'installation.

La version de l'image ISO IA doit être la même que la version du SE Oracle Solaris que vous prévoyez d'installer sur le client.

## Création d'un service d'installation IA

Créez un service d'installation pour associer une image d'installation à un service d'installation nommé. Les systèmes client utilisent le nom du service d'installation pour rechercher l'image d'installation appropriée.

Un serveur d'installation peut posséder plusieurs services d'installation. Chaque service d'installation est associé à une seule image d'initialisation. Pour installer des clients SPARC et x86, par exemple, vous avez besoin d'un seul service d'installation associé à une image d'initialisation SPARC et d'un second service d'installation associé à une image d'initialisation x86.

Utilisez la commande `installadm create-service` pour créer un service d'installation IA. Attribuez un nom au service, spécifiez le chemin d'accès à l'image ISO à utiliser et indiquez le chemin d'accès de l'emplacement où vous souhaitez que l'image ISO soit décompressée. L'emplacement de décompression de l'image ISO est également appelée l'image cible ou réseau.

Si vous n'indiquez pas de nom, le nom attribué au service d'installation est `_install_service_numéro_de_port`.

Si vous ne spécifiez pas le chemin d'accès à l'image ISO à utiliser, le chemin d'accès que vous spécifiez pour l'image réseau doit déjà contenir les fichiers d'image ISO décompressés.

La commande `installadm create-service` fournit également une image d'installation réseau sur un serveur Web en cours d'exécution sur le port 5555. Par exemple, l'adresse du serveur Web peut être `http://serveur.ai:5555/export/aiserver/s11-ai-x86/s11-x86`.

La commande `create-service` peut configurer Oracle Solaris DHCP sur le serveur d'installation IA comme indiqué dans [“Création d'un service d'installation comprenant la configuration du serveur DHCP Oracle Solaris” à la page 22](#). Reportez-vous au [Chapitre 6, “Configuration de DHCP pour le programme d'installation automatisée”](#) si vous souhaitez effectuer l'une des tâches DHCP suivantes :

- Configurez un serveur DHCP Oracle Solaris distinct.
- Configurez un serveur DHCP Oracle Solaris en vue d'une utilisation avec le programme d'installation automatisée.
- Configurez un serveur DHCP ISC.
- Configurez un serveur DHCP ISC en vue d'une utilisation avec le programme d'installation automatisée.

Le serveur DHCP doit être en mesure de fournir des informations DNS aux clients d'installation.

## Création d'un service d'installation sans configuration DHCP

L'exemple suivant crée un service d'installation IA pour les clients SPARC. Ce processus est le même pour un service d'installation x86, bien que la sortie soit différente. L'option `-n` spécifie le nom du service, et l'option `-s` spécifie le chemin d'accès au fichier image ISO IA utilisé pour créer ce service. Dans cet exemple, DHCP est déjà configuré sur un serveur différent ou le sera ultérieurement. Si la commande `create-service` ne détecte pas DHCP sur ce serveur d'installation IA, la sortie de la commande affiche des instructions pour la création d'une macro DHCP en utilisant `dhtadm(1M)` pour ajouter la macro à la table de configuration DHCP, `dhcptab(4)`. Reportez-vous au [Chapitre 6, “Configuration de DHCP pour le programme d'installation automatisée”](#) pour plus d'informations.

```
# installadm create-service -n s11-sparc \
-s /rpool/ai/s11_sparc/iso/s11-ai-sparc.iso \
/rpool/ai/s11_sparc/target
Setting up the target image at /rpool/ai/s11_sparc/target ...
Registering the service s11-sparc._OSInstall._tcp.local

Detected that DHCP is not set up on this server.
If not already configured, please create a DHCP macro
named dhcp_macro_s11-sparc with:
  Boot server IP (BootSrvA) : 10.6.68.29
  Boot file      (BootFile) : http://10.6.68.29:5555/cgi-bin/wanboot-cgi
If you are running the Solaris DHCP server, use the following
command to add the DHCP macro, dhcp_macro_s11-sparc:
  /usr/sbin/dhtadm -g -A -m dhcp_macro_s11-sparc \
-d :BootSrvA=10.6.68.29:BootFile=\http://10.6.68.29:5555/cgi-bin/wanboot-cgi\":
Note: Be sure to assign client IP address(es) if needed
(e.g., if running the Solaris DHCP server, run pntadm(1M)).
Service discovery fallback mechanism set up
Creating SPARC configuration file
```

Cette commande affiche le nom et les valeurs d'une macro, `dhcp_macro_s11-sparc`, que vous devez ajouter au serveur DHCP.

Si vous utilisez le serveur DHCP Oracle Solaris, vous pouvez créer la macro sur le serveur DHCP en exécutant les commandes `dhtadm`, indiquées dans la sortie ci-dessus, sur le serveur DHCP. Sur les systèmes qui prennent en charge les interfaces graphiques, vous pouvez utiliser le gestionnaire DHCP, `dhcpgmr(1M)`, au lieu des commandes `dhtadm`.

## Création d'un service d'installation comprenant la configuration du serveur DHCP Oracle Solaris

Vous pouvez utiliser la commande `installadm create-service` pour configurer un serveur DHCP Oracle Solaris sur le serveur d'installation IA. Ce processus est le même pour les services d'installation SPARC et x86, bien que la sortie soit différente. L'exemple suivant crée un service d'installation pour les clients x86 dans lequel le réseau est constitué d'un seul sous-réseau, et le serveur d'installation fonctionne également en tant que serveur DHCP pour le réseau, en utilisant DNS pour la résolution des noms d'hôte. Le programme d'installation automatisée crée une nouvelle macro DHCP nommée `dhcp_macro_s11-x86`. Ce service d'installation sert cinq adresses IP (-c), à partir de 172.1.0.10 (-i). Reportez-vous à la section "[Configuration d'un serveur DHCP existant pour le programme d'installation automatisée](#)" à la page 84 pour plus d'informations.

```
# installadm create-service -n s11-x86 -i 172.1.0.10 -c 5 \
-s /rpool/ai/s11_x86/iso/s11-ai-x86.iso \
/rpool/ai/s11_x86/target
Setting up the target image at /rpool/ai/s11_x86/target ...
Registering the service s11_x86._OSInstall._tcp.local
Creating DHCP Server
Created DHCP configuration file.
Created dhcptab.
Added "Locale" macro to dhcptab.
Added server macro to dhcptab - line1-x4100.
DHCP server started.
Added network macro to dhcptab - 10.0.0.0.
Created network table.
Copying boot file to /tftpboot/pxegrub.I86PC.Solaris-1
Service discovery fallback mechanism set up
```

Dans cet exemple, vous pouvez passer en revue le fichier `menu.lst` dans `/tftpboot/menu.lst.s11_x86`.

Si le service DHCP est déjà configuré sur ce serveur, vous pouvez utiliser les options `-i` et `-c` pour mettre à jour le serveur DHCP avec de nouvelles adresses IP pour le service IA nommé.

Vous pouvez visualiser les résultats de la configuration DHCP dans la table correspondante à l'aide de l'utilitaire DHCP Manager `dhcpgmr(1M)`.

## Consultez les instructions d'installation par défaut

Un fichier manifest IA contient les instructions d'installation et de configuration qui peuvent être utilisées pour un ou plusieurs clients. Chaque image d'initialisation par défaut comprend un fichier manifest IA qui peut être utilisé pour les clients de n'importe quel service d'installation créé à l'aide de cette image d'initialisation. Le fichier manifest est décompressé ainsi que les autres fichiers de l'image. Si, par exemple, le chemin de l'image réseau d'un service est `/rpool/ai/s11_sparc/target`, le fichier manifest IA par défaut est dans `/rpool/ai/s11_sparc/target/auto_install/default.xml`.

Il s'agit du fichier manifest IA par défaut pour tous les clients qui utilisent tout service d'installation utilisant cette image d'installation.

---

**Remarque** – Ce fichier `default.xml` ne doit pas être supprimé, ni déplacé ou renommé.

---

Vérifiez ce fichier manifest par défaut pour déterminer s'il répond aux besoins de tous les clients qui vont utiliser un service d'installation basé sur cette image. “[Fichier manifest IA par défaut](#)” à la [page 37](#) affiche une copie du fichier manifest IA par défaut. Le fichier manifest par défaut peut être légèrement différent dans différentes images d'installation.

Pour effectuer différentes installations sur différents clients à l'aide de la même image d'installation, fournissez des fichiers manifest IA pour ce service d'installation. Plusieurs fichiers manifest IA peuvent être associés à chaque service d'installation. Les clients qui ne correspondent pas aux critères spécifiés pour utiliser n'importe quel fichier manifest personnalisé sont installés à l'aide des instructions figurant dans le fichier manifest IA par défaut.

---

**Remarque** – En général, vous ne devez pas modifier le fichier `chemin_image_installation_reseau/auto_install/default.xml`. Le fichier `chemin_image_installation_reseau/auto_install/default.xml` sera le fichier manifest IA par défaut de n'importe quel service d'installation créé à l'avenir utilisant cette image d'installation. Le fichier manifest IA par défaut doit fonctionner pour tout client ne correspondant pas à un fichier manifest personnalisé, pour n'importe quel service basé sur cette image.

---

Au lieu de modifier `chemin_image_installation_reseau/auto_install/default.xml`, prenez en compte les options suivantes :

- Ajoutez des fichiers manifest personnalisés aux services d'installation spécifiés.  
Pour créer des fichiers manifest d'installation personnalisés et les associer aux clients ou types de clients spécifiés, reportez-vous au [Chapitre 3, “Personnalisation des installations”](#).
- Modifiez le fichier manifest IA par défaut pour un service d'installation spécifié.

1. Copiez `chemin_image_installation_reseau/auto_install/default.xml` dans un nouvel emplacement : `nouvel_emplacement/default.xml`.
2. Modifiez `nouvel_emplacement/default.xml`.  
Pour modifier les spécifications d'installation telles que le disque cible ou les packages supplémentaires à installer, consultez le [Chapitre 4, “Spécification des instructions d'installation”](#). Pour changer les spécifications de configuration telles que le compte d'utilisateur ou le mot de passe root, reportez-vous au [Chapitre 5, “Configuration du système client”](#).
3. Ajoutez le fichier manifest par défaut modifié à un service d'installation spécifié.

Le service d'installation utilise une copie interne du fichier manifest par défaut qui était dans l'image lors de la création du service. La modification du fichier manifest IA par défaut se trouvant dans le répertoire de l'image ne modifie pas le fichier manifest IA par défaut pour tous les services qui ont déjà été créés. Pour modifier la copie interne du fichier manifest IA par défaut pour un service existant, ajoutez le fichier manifest modifié au service.

Utilisez la commande `installadm add-manifest` pour ajouter `nouvel_emplacement/default.xml` à un service d'installation spécifié. Reportez-vous à la section “Ajout d'un fichier manifest IA ” à la page 26.

- Le nouveau fichier manifest par défaut doit être nommé `default.xml`.
- La valeur de l'attribut `name` de la balise `<ai_instance>` doit être `default`.
- Ne spécifiez aucun critère de client dans la commande `add-manifest`.

## Gestion d'un serveur d'installation

Une fois que vous avez configuré un serveur d'installation IA, vous pouvez être amené à effectuer les tâches suivantes : Consultez également la page de manuel `installadm(1M)`.

- “Ajout ou suppression d'un service d'installation” à la page 25
- “Activation ou désactivation d'un service d'installation” à la page 25
- “Association de clients à des services d'installation” à la page 25
- “Association d'instructions d'installation spécifiques au client à des services d'installation” à la page 26
- “Affichage de tous les services d'installation du serveur d'installation” à la page 29
- “Affichage des clients associés aux services d'installation” à la page 30
- “Affichage de tous les fichiers manifest” à la page 30



## Ajout ou suppression d'un service d'installation

Vous avez besoin d'un service d'installation pour chaque image d'initialisation que vous souhaitez utiliser. Vous pouvez créer plusieurs services d'installation associés à la même image d'initialisation. Pour créer un service d'installation, utilisez la commande `installadm create-service` comme décrit dans la section [“Création d'un service d'installation IA”](#) à la page 20.

Utilisez la commande suivante pour supprimer un service d'installation :

```
# installadm delete-service [-x] svcname
```

*svcname* est le nom du service à supprimer. Si vous spécifiez l'option `-x`, l'image d'installation associée est également supprimée.

Lorsque vous supprimez un service d'installation, les macros DHCP associées restent dans la table DHCP. La sortie de la commande `installadm delete-service` vous indique les macros DHCP à supprimer.

## Activation ou désactivation d'un service d'installation

Utilisez la commande suivante pour activer un service d'installation.

```
# installadm enable svcname
```

*svcname* est le nom du service à activer. Cette commande active également le serveur Web associé au service.

Utilisez la commande suivante pour désactiver un service d'installation.

```
# installadm disable svcname
```

Cette commande désactive également le serveur Web associé au service.

## Association de clients à des services d'installation

La commande `installadm create-client` indique exactement au client quel service d'installation utiliser. Reportez-vous à la section [“Configuration d'un client d'installation”](#) à la page 92 pour plus d'exemples, notamment de sortie.

## Ajout d'un client à un service d'installation

Utilisez la commande `installadm create-client` pour associer un client à un service d'installation. La commande suivante permet d'ajouter le client avec l'adresse MAC `00:14:4f:A7:65:70` au service d'installation `s11-sparc`.

```
# installadm create-client -e 00:14:4f:a7:65:70 -n s11-sparc
```

L'exemple suivant ajoute un client x86 et redirige la sortie d'installation vers une console série.

```
# installadm create-client -e c0ffec0ffee -n s11-x86 -b 'console=ttya'
```

Un client peut être associé à un seul service d'installation. Si vous exécutez la commande `installadm create-client` plusieurs fois et indiquez la même adresse MAC à chaque fois, ce client est uniquement associé au dernier service d'installation spécifié.

## Suppression d'un client d'un service d'installation

Utilisez la commande `installadm delete-client` pour supprimer un client d'un service d'installation. La commande suivante permet de supprimer le client avec l'adresse MAC `00:14:4f:A7:65:70`. Vous n'avez pas besoin de spécifier le nom de service car un client ne peut être associé qu'à un seul service d'installation.

```
# installadm delete-client -e 00:14:4f:a7:65:70
```

## Association d'instructions d'installation spécifiques au client à des services d'installation

Vous pouvez spécifier plusieurs ensembles d'instructions d'installation et de configuration système pour chaque service d'installation. Vous pouvez également indiquer quel ensemble d'instructions doit être utilisé pour chaque client. Toutes ces informations sont associées au service d'installation.

## Ajout d'un fichier manifest IA

Utilisez la commande `installadm add-manifest` pour ajouter un fichier manifest IA personnalisé à un service d'installation.

```
# installadm add-manifest -m manifest -n service_name \  
[-c criteria="value"|"range" ... \  
| -C criteria_file]
```

La valeur de *manifest* est un chemin d'accès complet et un nom de fichier dont l'extension est `.xml`. Le fichier *manifest* contient un fichier manifest IA (instructions d'installation). Le fichier manifest peut également faire référence ou incorporer un fichier manifest SC (instructions de configuration système).

Pour spécifier des critères pour déterminer quels clients doivent utiliser les instructions figurant dans le fichier manifest spécifié, utilisez l'option `-c` ou `-C`. La valeur de *fichier\_de\_critères* est un chemin d'accès complet et un nom de fichier. Des exemples de fichiers de critères sont présentés ci-dessous et dans le [Chapitre 3, "Personnalisation des installations"](#).

- Si vous remplacez le fichier manifest `default.xml` d'un service, n'indiquez pas de critères de client. Le fichier manifest IA par défaut permet d'installer tous les clients ne correspondant à aucun fichier manifest personnalisé. Reportez-vous à la section "[Consultez les instructions d'installation par défaut](#)" à la page 23.
- Pour tous les autres fichiers manifest IA que vous ajoutez, l'option `-c` ou `-C` est requise. Seul le fichier manifest IA par défaut n'est associé à aucun critère de client. Tous les autres fichiers manifest IA sont des fichiers manifest IA personnalisés pour les clients ou groupes de clients spécifiés.

La commande suivante ajoute le fichier manifest `manifest_t200.xml` au service d'installation `s11-sparc`. L'option `-c` spécifie que tous les clients qui utilisent ce service d'installation et s'identifient en tant que serveurs Sun Fire T200 se voient affecter les instructions d'installation et de configuration `manifest_t200.xml`.

```
# installadm add-manifest -m /rpool/ai/s11_sparc/manifests/manifest_t200.xml \
-n s11-sparc -c platform="SUNW,Sun-Fire-T200"
```

La commande suivante est l'équivalent de la commande précédente si le contenu du fichier `criteria_t200.xml` est comme indiqué.

```
# installadm add-manifest -m /rpool/ai/s11_sparc/manifests/manifest_t200.xml \
-n s11-sparc -C /rpool/ai/s11_sparc/manifests/criteria_t200.xml
```

Vous trouverez ci-dessous le contenu du fichier `criteria_t200.xml`.

```
<ai_criteria_manifest>
  <ai_criteria name="platform">
    <value>SUNW,Sun-Fire-T200</value>
  </ai_criteria>
</ai_criteria_manifest>
```

La commande `installadm add-manifest` vérifie que les critères du même type ne se chevauchent pas. Par exemple, si une spécification de critère correspond aux adresses IP de 10.0.0.0 à 10.255.255.255, `installadm` s'arrête et un message d'erreur s'affiche si vous tentez d'ajouter une spécification de critère correspondant à l'adresse IP 10.10.10.10. Pour plus d'informations sur les spécifications de critères, reportez-vous au [Chapitre 3, "Personnalisation des installations"](#).

## Modification des critères pour un fichier manifest d'instructions d'installation

Utilisez la commande `installadm set-criteria` pour mettre à jour les critères du client associés au fichier manifest IA que vous avez déjà ajouté à un service à l'aide de la commande `installadm add-manifest`.

- Utilisez `installadm set-criteria` avec l'option `-a` afin d'ajouter plusieurs critères pour l'utilisation du fichier manifest spécifié. Les critères actuels sont conservés, et les nouveaux critères sont ajoutés.
- Utilisez `installadm set-criteria` avec l'option `-c` ou `-C` afin de remplacer les critères pour l'utilisation du fichier manifest spécifié. Les critères actuels sont supprimés et remplacés par les nouveaux critères.

```
# installadm set-criteria -m manifest -n service_name \
-a criteria="value" | "range" ... \
| -c criteria="value"|"range" ... \
| -C criteria.xml
```

La valeur de *manifest* est le nom du fichier manifest que la commande `installadm list -m` renvoie. Reportez-vous à la section “Affichage de tous les fichiers manifest” à la page 30.

La commande suivante ajoute des critères au fichier manifest `manifest_t200.xml` indiqué dans la section “Ajout d'un fichier manifest IA” à la page 26.

```
# installadm set-criteria -m manifest_t200.xml -n s11-sparc -a mem="4096-unbounded"
```

Le résultat des deux commandes est que le fichier manifest `manifest_t200.xml` est utilisé par n'importe quel client utilisant ce service d'installation, étant un serveur Sun Fire T200 Server et disposant d'au moins 4 Go de mémoire.

Vous pouvez obtenir le même résultat en utilisant l'option `-C` au lieu de l'option `-a` avec le fichier `criteria_t200.xml` suivant.

```
<ai_criteria_manifest>
  <ai_criteria name="platform">
    <value>SUNW,Sun-Fire-T200</value>
  </ai_criteria>
  <ai_criteria name="mem">
    <range>
      4096
      unbounded
    </range>
  </ai_criteria>
</ai_criteria_manifest>
```

## Suppression d'un fichier manifest d'instructions d'installation

Utilisez la commande `installadm delete-manifest` pour supprimer un fichier manifest IA à partir d'un service d'installation. La valeur de *manifest* est le nom de fichier manifest renvoyé par la commande `installadm list -m`. Reportez-vous à la section “Affichage de tous les fichiers manifest” à la page 30.

Vous ne pouvez pas supprimer le fichier manifest IA par défaut.

La commande suivante supprime le fichier manifest IA `manifest_mac1.xml` du service d'installation `s11-sparc`.

```
# installadm delete-manifest -m manifest_mac1.xml -n s11-sparc
```

## Affichage des informations relatives aux services d'installation

Utilisez la commande `installadm list` pour afficher les informations relatives aux services d'installation.

### Affichage de tous les services d'installation du serveur d'installation

La commande suivante affiche tous les services d'installation de ce serveur. Dans cet exemple, deux services d'installation activés ont été trouvés. Les services désactivés possèdent la valeur d'état `off`.

```
# installadm list
```

Service Name	Status	Arch	Port	Image Path
s11-sparc	on	sparc	46501	/rpool/ai/s11_sparc/target
s11-x86	on	x86	46502	/rpool/ai/s11_x86/target

### Affichage des informations relatives à un service d'installation spécifié

La commande suivante affiche les informations sur le service d'installation spécifié par l'option `-n`:

```
# installadm list -n s11-sparc
```

Service Name	Status	Arch	Port	Image Path
s11-sparc	on	sparc	46501	/rpool/ai/s11_sparc/target

## Affichage des clients associés aux services d'installation

La commande suivante répertorie tous les clients qui sont associés aux services d'installation de ce serveur d'installation. Les clients ont été associés aux services d'installation à l'aide de la commande `installadm create-client`. Reportez-vous à la section [“Ajout d'un client à un service d'installation”](#) à la page 26.

```
# installadm list -c
```

Service Name	Client Address	Arch	Image Path
s11-sparc	00:14:4F:A7:65:70	sparc	/rpool/ai/s11_sparc/target
s11-x86	08:00:27:8B:BD:71	x86	/rpool/ai/s11_x86/target
	01:C2:52:E6:4B:E0	x86	/rpool/ai/s11_x86/target

## Affichage des clients associés à un service d'installation spécifique

La commande suivante répertorie tous les clients qui ont été ajoutés au service d'installation spécifié. Dans l'exemple suivant, un client est associé au service d'installation `s11-sparc`.

```
# installadm list -c -n s11-sparc
```

Service Name	Client Address	Arch	Image Path
s11-sparc	00:14:4f:a7:65:70	sparc	/rpool/ai/s11_sparc/target

## Affichage des informations sur les installations personnalisées

Les commandes de cette section indiquent quels fichiers manifest IA sont associés à un service d'installation spécifique. Ces commandes indiquent également les critères de client qui sont associés à chaque fichier manifest IA. Reportez-vous aux commandes `installadm add-manifest` et `installadm set-criteria` dans [“Association d'instructions d'installation spécifiques au client à des services d'installation”](#) à la page 26.

## Affichage de tous les fichiers manifest

La commande suivante répertorie tous les fichiers manifest IA personnalisés pour tous les services d'installation de ce serveur d'installation. Le fichier manifest IA par défaut n'est pas répertorié. Si un service d'installation n'est associé à aucun fichier manifest IA personnalisé, le service n'apparaît pas dans cette liste.

La colonne Manifest affiche le nom du fichier manifest à partir de l'attribut `name` de la balise `<ai_instance>`. Dans cet exemple, tous les fichiers manifest IA personnalisés sont affichés pour deux services d'installation, `s11-sparc` et `s11-x86`.

```
# installadm list -m

Service Name Manifest
-----
s11-sparc    manifest_mac1.xml
             manifest_t200.xml
s11-x86      manifest_mac2.xml
             manifest_mac3.xml
             manifest_ipv41.xml
             manifest_ipv42.xml
             manifest_mem1.xml
```

## Affichage des fichiers manifest associés à un service d'installation spécifié

L'exemple suivant montre tous les fichiers manifest IA personnalisés associés au service d'installation s11-x86. La colonne Manifest affiche le nom du fichier manifest à partir de l'attribut name de la balise <ai\_instance>. La colonne Criteria affiche les critères de client qui sont associés à chaque fichier manifest.

```
# installadm list -m -n s11-x86

Manifest          Criteria
-----
manifest_mac2.xml arch = i86pc
                  mac  = 01:C2:52:E6:4B:E0

manifest_mac3.xml arch = i86pc
                  mac  = 01:C2:52:E6:4B:E6 - 01:C2:52:E6:4B:E9

manifest_ipv41.xml arch = i86pc
                  ipv4 = 192.168.168.12

manifest_ipv42.xml arch = i86pc
                  ipv4 = 192.168.168.251

manifest_mem1.xml arch = i86pc
                  mem  = 2048 MB
```

## Administration du service SMF IA

Sur le serveur IA, le service SMF `svc:/system/install/server:default` est le service qui représente l'état général de l'application de serveur IA et de tous les services d'installation. Ce service SMF contient les données propres à chaque service d'installation.

**EXEMPLE 2-1** Activation du service SMF IA

Le service SMF IA est activé lors de l'exécution de la commande `installadm create-service`. Le service SMF IA est également activé lorsque vous exécutez toute autre commande `installadm` affectant les services d'installation existants. Pour activer manuellement le service SMF IA, exécutez la commande suivante :

```
# svcadm enable svc:/system/install/server:default
```

Le service passe en mode de maintenance si aucun service d'installation n'est actuellement activé sur le serveur d'installation.

**EXEMPLE 2-2** Désactivation du service SMF IA

Pour désactiver le service SMF IA, exécutez la commande suivante :

```
# svcadm disable svc:/system/install/server:default
```

Ne désactivez pas le service SMF IA si tout service d'installation IA est toujours activé. Reportez-vous à la section [“Affichage de tous les services d'installation du serveur d'installation”](#) à la page 29 pour plus d'informations sur la vérification de l'activation des services d'installation.



## Personnalisation des installations

---

Pour personnaliser une installation, vous pouvez personnaliser les instructions d'installation et les instructions de configuration du système. Spécifiez ensuite les critères de client pour faire correspondre l'installation personnalisée et les instructions de configuration avec les clients qui répondent aux critères spécifiés. Les instructions personnalisées sont utilisées à la place des instructions par défaut pour installer le client.

Un service IA possède un ou plusieurs fichiers manifest IA qui contiennent des instructions d'installation et de configuration. Vous pouvez préciser des caractéristiques de clients comme le type de plate-forme ou la quantité de mémoire, afin que les clients qui correspondent aux critères spécifiés utilisent le fichier manifest IA associé pour terminer leur installation.

### Mise en correspondance des instructions d'installation avec les caractéristiques de clients

Lorsque vous utilisez le programme d'installation automatisée, vous devez d'abord configurer un serveur DHCP et un serveur IA. Le serveur IA dispose d'au moins une image d'initialisation IA et d'un service IA qui est associé à cette image d'initialisation. Lors de l'initialisation d'un client, DHCP dirige le client sur le serveur d'installation. Le client est associé à un service d'installation donné. Le service d'installation utilise les méthodes décrites dans ce chapitre pour faire correspondre le client avec les instructions d'installation et de configuration correctes à utiliser.

Pour définir les installations qui utilisent des images IA différentes (une image SPARC et une image x86, ou des versions d'Oracle Solaris différentes), créez un service distinct pour chaque image. Reportez-vous à la section [“Configuration de l'image d'installation et du service d'installation”](#) à la page 17.

Pour associer un client à un service d'installation spécifique, ajoutez ce client au service d'installation. Reportez-vous au [Chapitre 7, “Installation de systèmes clients”](#). Spécifiez l'adresse

MAC du client et le nom du service d'installation à utiliser pour ce client. Lorsque le client avec cette adresse MAC s'initialise, DHCP dirige le client sur le serveur d'installation et le client utilise le service d'installation spécifié.

Pour définir plusieurs types d'installation pour une image IA, créez d'autres fichiers manifest IA et ajoutez-les au service IA pour cette image. Spécifiez les critères qui définissent quel clients doivent utiliser quels fichiers manifest IA. Reportez-vous à la section [“Association d'instructions d'installation spécifiques au client à des services d'installation”](#) à la page 26.

Pour créer un fichier manifest IA personnalisé, commencez par une copie du fichier manifest par défaut `chemin_image_installation_réseau/auto_install/default.xml` pour le service. Reportez-vous également aux sections [“Fichier manifest IA par défaut”](#) à la page 37 et [“Fichier manifest IA annoté”](#) à la page 67 pour obtenir des exemples. Pour modifier les spécifications d'installation telles que le disque cible ou les packages supplémentaires à installer, consultez le [Chapitre 4, “Spécification des instructions d'installation”](#). Pour changer les spécifications de configuration telles que le compte d'utilisateur ou le mot de passe root, reportez-vous au [Chapitre 5, “Configuration du système client”](#).

Un fichier manifest IA est sélectionné pour un client en fonction de l'algorithme suivant :

- Si aucun fichier manifest IA personnalisé n'est défini pour ce service d'installation, le fichier manifest IA par défaut est utilisé. Le fichier manifest IA par défaut n'est associé à aucun critère de client. Reportez-vous à la section [“Consultez les instructions d'installation par défaut”](#) à la page 23.
- Si des fichiers manifest IA personnalisés sont définis pour ce service d'installation, mais que le client ne correspond aux critères d'aucun fichier manifest IA personnalisé, le client utilise le fichier manifest IA par défaut.
- Si le client correspond aux critères qui ont été spécifiés pour un fichier manifest IA personnalisé, il utilise ce fichier.

Si les caractéristiques du client correspondent à plusieurs fichiers manifest IA, elles sont évaluées dans l'ordre indiqué dans le tableau ci-dessous pour sélectionner un fichier manifest pour l'installation. L'outil `instadm` vérifie que les critères de même type ne se chevauchent pas. Reportez-vous à la section [“Ajout d'un fichier manifest IA”](#) à la page 26.

Plusieurs critères ne se chevauchant pas sont utilisés dans l'ordre indiqué dans le tableau ci-après. Par exemple, si une seule spécification de critère correspond à l'adresse MAC du client et qu'une autre spécification de critère correspond à l'adresse IP du même client, le fichier manifest associé à la spécification de critère d'adresse MAC est utilisé, car `mac` a une priorité plus élevée pour la sélection qu'`ip4`.

TABLEAU 3-1 Balises et hiérarchie des critères

Nom de critère	Description	Exemples
arch	uname -m	i86pc ou sun4u ou sun4v  <pre>&lt;ai_criteria name="arch"&gt;   &lt;value&gt;i86pc&lt;/value&gt; &lt;/ai_criteria&gt;</pre>
mac	Adresse MAC ou plage d'adresses MAC	<pre>&lt;ai_criteria name="mac"&gt;   &lt;value&gt;0:14:4F:20:53:97&lt;/value&gt; &lt;/ai_criteria&gt;</pre> <pre>&lt;ai_criteria name="mac"&gt;   &lt;range&gt;     0:14:4F:20:53:94     0:14:4F:20:53:A0   &lt;/range&gt; &lt;/ai_criteria&gt;</pre>
ipv4	Adresse IP ou plage d'adresses IP	<pre>&lt;ai_criteria name="ipv4"&gt;   &lt;value&gt;10.6.68.127&lt;/value&gt; &lt;/ai_criteria&gt;</pre> <pre>&lt;ai_criteria name="ipv4"&gt;   &lt;range&gt;     10.6.88.1     10.6.68.200   &lt;/range&gt; &lt;/ai_criteria&gt;</pre>
cpu	uname -p	i386 ou sparc  <pre>&lt;ai_criteria name="cpu"&gt;   &lt;value&gt;sparc&lt;/value&gt; &lt;/ai_criteria&gt;</pre>
platform	uname -i	i86pc ou SUNW, Sun-Fire-T200 ou SUNW, Sun-Fire-880  <pre>&lt;ai_criteria name="platform"&gt;   &lt;value&gt;SUNW, Sun-Fire-T200&lt;/value&gt; &lt;/ai_criteria&gt;</pre>

**TABEAU 3-1** Balises et hiérarchie des critères (Suite)

Nom de critère	Description	Exemples
mem	Taille de la mémoire en méga-octets, ou une plage de tailles de mémoire	<p>Cet exemple correspond aux clients disposant d'au moins 2 Go de mémoire. Le mot-clé <code>unbounded</code> indique qu'il n'existe aucune limite supérieure dans une plage.</p> <pre>&lt;ai_criteria name="mem"&gt;   &lt;range&gt;     2048     unbounded   &lt;/range&gt; &lt;/ai_criteria&gt;</pre>

**EXEMPLE 3-1** Mise en correspondance de clients avec des fichiers manifest IA

Dans l'exemple suivant, deux fichiers manifest IA personnalisés ont été ajoutés au même service d'installation. Les critères de client associés à ces fichiers manifest sont comme indiqué.

Le fichier manifest IA `manifest_x86.xml` a été ajouté au service avec le fichier de critères suivant qui spécifie l'architecture client :

```
<ai_criteria_manifest>
  <ai_criteria name="arch">
    <value>i86pc</value>
  </ai_criteria>
</ai_criteria_manifest>
```

Le fichier manifest IA `manifest_mac1.xml` a été ajouté au service avec le fichier de critères suivant qui spécifie une adresse MAC de client :

```
<ai_criteria_manifest>
  <ai_criteria name="mac">
    <value>00:14:4f:a7:65:70</value>
  </ai_criteria>
</ai_criteria_manifest>
```

Si un client x86 est en cours d'installation, le fichier `manifest_x86.xml` lui est attribué.

Si un client SPARC avec l'adresse MAC `00:14:4f:A7:65:70` est en cours d'installation, le fichier `manifest_mac1.xml` lui est attribué.

Si un système SPARC avec une autre adresse MAC est en cours d'installation, le fichier manifest IA par défaut lui est attribué.

## Fichier manifest IA par défaut

Lorsque vous créez un service d'installation, `chemin_image_service_installation/auto_install/default.xml` est le fichier manifest IA par défaut pour tous les clients qui utilisent ce service.

Le fichier manifest IA par défaut est illustré ci-dessous. Ce fichier manifest IA par défaut peut être légèrement différent dans différentes images d'installation.

- Le fichier manifest par défaut ne contient aucune section de périphérique cible. La section “Définition d'une cible pour l'installation” à la page 45 décrit la détermination de l'emplacement cible par défaut pour l'installation.
- Les instructions d'installation de logiciel spécifient le référentiel de packages IPS par défaut et installent les deux packages qui sont requis pour chaque installation.
- Les instructions `add_drivers` recherchent dans chaque client les pilotes de périphériques manquants. Tous les pilotes qui ont été trouvés dans le référentiel de packages IPS par défaut et ne sont pas identifiés comme des pilotes tiers sont installés. Pour installer tout pilote tiers nécessaire, spécifiez `<search_all addall="true"/>`.
- Les instructions de configuration du système sont également appelées fichier manifest SC. Le fichier manifest SC intégré est contenu dans les balises `<sc_embedded_manifest>` et `</sc_embedded_manifest>`. Le fichier manifest SC indique comment configurer le client une fois l'installation terminée. Le fichier manifest SC définit les propriétés des services SMF. Notez que l'ensemble du fichier manifest SC se trouve à l'intérieur d'un commentaire.  
`jack` est le nom d'utilisateur par défaut et le mot de passe utilisateur par défaut. Le mot de passe `root` par défaut est `solaris`.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Copyright (c) 2008, 2010, Oracle and/or its affiliates. All rights reserved.

-->
<!DOCTYPE auto_install SYSTEM "file:///usr/share/auto_install/ai.dtd">
<auto_install>
  <ai_instance name="default">
    <software>
      <source>
        <publisher name="solaris">
          <origin name="http://pkg.oracle.com/solaris/release"/>
        </publisher>
      </source>
    </software>
    <!--
      By default the latest build available, in the specified IPS
      repository, is installed. If another build is required, the
      build number has to be appended to the 'entire' package in following
      form:

      <name>pkg:/entire@0.5.11-0.build#</name>
    -->
    <software_data action="install" type="IPS">
```

```

    <name>pkg:/entire</name>
    <name>pkg:/babel_install</name>
    <!--
    The following packages are required by iSCSI and included
    by default to make it easier for users to enable iSCSI if
    desired. They can be deleted from this list if iSCSI isn't
    used. See iscsiadm(1m) man page for more information.
    support for iSCSI.
-->
-->
    <name>pkg:/network/iscsi/initiator</name>
    <name>pkg:/network/iscsi/iser</name>
</software_data>
<!--
    babel_install and slim_install are group packages used to
    define the default installation. They are removed here so
    that they do not inhibit removal of other packages on the
    installed system.
-->
    <software_data action="uninstall" type="IPS">
        <name>pkg:/babel_install</name>
        <name>pkg:/slim_install</name>
    </software_data>
</software>
<!--
Add missing driver packages to a booted install image so an
installation can complete. Add packages to target as well.
<search_all> searches and installs from configured repo.
-->
<add_drivers>
    <search_all/>
</add_drivers>
<sc_embedded_manifest name="AI">
    <!-- <?xml version='1.0'?>
    <!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
    <service_bundle type="profile" name="system configuration">
        <service name="system/install/config" version="1" type="service">
            <instance name="default" enabled="true">
                <property_group name="user_account" type="application">
                    <propval name="login" type="astring" value="jack"/>
                    <propval name="password" type="astring" value="encrypted_password"/>
                    <propval name="description" type="astring" value="default user"/>
                    <propval name="shell" type="astring" value="/usr/bin/bash"/>
                    <propval name="uid" type='count' value='101'/>
                    <propval name="gid" type='count' value='10'/>
                    <propval name="type" type="astring" value="normal"/>
                    <propval name="roles" type="astring" value="root"/>
                </property_group>

                <property_group name="root_account" type="application">
                    <propval name="password" type="astring" value="encrypted_password"/>
                    <propval name="type" type="astring" value="role"/>
                </property_group>

                <property_group name="other_sc_params" type="application">
                    <propval name="timezone" type="astring" value="GMT"/>
                    <propval name="hostname" type="astring" value="solaris"/>
                </property_group>
            </instance>
        </service>

```

```
<service name="system/console-login" version="1" type="service">
  <property_group name="ttymon" type="application">
    <propval name="terminal_type" type="astring" value="sun"/>
  </property_group>
</service>

<service name='system/keymap' version='1' type='service'>
  <instance name='default' enabled='true'>
    <property_group name='keymap' type='system'>
      <propval name='layout' type='astring' value='US-English' />
    </property_group>
  </instance>
</service>

<service name="network/physical" version="1" type="service">
  <instance name="nwam" enabled="true"/>
  <instance name="default" enabled="false"/>
</service>
</service_bundle>
-->
</sc_embedded_manifest>
</ai_instance>
</auto_install>
```





# Spécification des instructions d'installation

---

Lorsque vous créez un service d'installation IA, vous obtenez un fichier manifest IA par défaut expliquant l'installation des clients. Reportez-vous à la section “[Consultez les instructions d'installation par défaut](#)” à la page 23 pour plus d'informations sur le fichier manifest IA par défaut et comment personnaliser le fichier manifest par défaut.

Ce chapitre explique la création de fichiers manifest IA personnalisés pour des clients particuliers.

## Balises de fichier manifest IA

Le fichier manifest IA est un fichier XML constitué d'éléments de balises, d'attributs et de valeurs qui définissent le mode d'installation d'un client.

### EXEMPLE 4-1 Modèle du fichier manifest IA

Cet exemple montre l'ensemble des balises requises pour chaque fichier manifest IA. Beaucoup d'autres balises sont disponibles, mais elles ne sont pas requises. Par exemple, ce modèle n'affiche aucune balise pour la définition d'une cible d'installation ou pour l'installation automatique de pilotes manquants. La plupart des balises disponibles pour un fichier manifest IA sont présentées dans le [Tableau 4-1](#) et abordées dans ce chapitre.

- Vous devez donner un nom unique au fichier manifest.
- Vous devez indiquer au moins un référentiel de packages IPS.
- Tout fichier manifest IA doit installer au moins les packages `entire` et `babel_install`.
- L'action `uninstall` n'est pas nécessaire. Toutefois, il est fortement recommandé de laisser chaque fichier manifest IA désinstaller les packages `babel_install` et `slim_install`. Pour obtenir une explication sur la raison de la désinstallation de ces packages, voir [Exemple 4-21](#).

```
<auto_install>  
  <ai_instance name="AI_manifest_name">
```

## EXEMPLE 4-1 Modèle du fichier manifest IA (Suite)

```

<software>
  <source>
    <publisher name="publisher_name">
      <origin name="repository_URI"/>
    </publisher>
  </source>
  <software_data action="install" type="IPS">
    <name>pkg:/entire</name>
    <name>pkg:/babel_install</name>
  </software_data>
  <software_data action="uninstall" type="IPS">
    <name>pkg:/babel_install</name>
    <name>pkg:/slim_install</name>
  </software_data>
</software>
</ai_instance>
</auto_install>

```

Le tableau suivant répertorie les balises les plus couramment utilisées dans les fichiers manifest IA. Le tableau vous indique où trouver des informations à propos de l'utilisation des balises.

TABLEAU 4-1 Mappe de balises de fichier manifest IA

Éléments et attributs de balise	Référence
<pre> &lt;auto_install&gt;   &lt;ai_instance name="" http_proxy="" auto_reboot=""&gt; </pre>	<a href="#">“Création d'un fichier manifest IA personnalisé” à la page 44</a>
<pre> &lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;disk_name name="" name_type=""&gt; </pre>	<a href="#">“Spécifications déterministes de disque cible” à la page 47 dans “Définition d'une cible pour l'installation” à la page 45</a>
<pre> &lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;iscsi name="" source="" target_lun="" target_port=""&gt;       &lt;ip&gt; </pre>	Exemple 4-6
<pre> &lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;disk_keyword key="boot_disk"/&gt; </pre>	Exemple 4-7
<pre> &lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;disk_prop dev_type="" dev_vendor="" dev_size=""&gt; </pre>	<a href="#">“Spécifications non déterministes de disque cible” à la page 50</a>

TABLEAU 4-1 Mappe de balises de fichier manifest IA (Suite)

Éléments et attributs de balise	Référence
<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;partition action=""&gt;       &lt;partition name="" part_type=""&gt;       &lt;size start_sector="" val=""/&gt;</pre>	<p>“Configuration du partitionnement sur un client x86” à la page 51</p>
<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;slice action=""&gt;       &lt;slice name="" is_root="" force=""&gt;       &lt;size val=""/&gt;</pre>	<p>“Configuration de tranches sur un disque” à la page 55</p>
<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;swap&gt;       &lt;zvol action="" name=""&gt;       &lt;size val=""/&gt;</pre>	<p>“Configuration du swap et du vidage sur le périphérique d’installation” à la page 57</p>
<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;dump&gt;       &lt;zvol action="" name=""&gt;       &lt;size val=""/&gt;</pre>	<p>“Configuration du swap et du vidage sur le périphérique d’installation” à la page 57</p>
<pre>&lt;software name=""&gt;   &lt;source&gt;     &lt;publisher name=""&gt;     &lt;origin name=""/&gt;</pre>	<p>“Spécification d’une source de packages à installer” à la page 59 dans “Installation de logiciels” à la page 59</p>
<pre>&lt;software name=""&gt;   &lt;software_data action="" type=""&gt;   &lt;name&gt;</pre>	<p>“Spécification des packages à installer” à la page 61</p>
<pre>&lt;add_drivers&gt;   &lt;search_all addall=""&gt;   &lt;source&gt;     &lt;publisher name=""&gt;     &lt;origin name=""/&gt;</pre>	<p>“Identification et installation de pilotes manquants sur une cible d’installation” à la page 63</p>

## Création d'un fichier manifest IA personnalisé

Pour créer et appliquer un fichier manifest IA personnalisé, suivez les étapes ci-après :

1. Copiez un fichier manifest IA existant. Une copie du fichier manifest IA par défaut obtenue lors de la création d'un service d'installation se trouve à `chemin_image_service_installation/auto_install/default.xml`.  
Copiez ce fichier manifest par défaut ou un autre fichier manifest IA sous un nouveau nom de fichier.
2. Modifiez le nouveau fichier, en ajoutant des balises et des valeurs en fonction des informations figurant dans ce chapitre.
3. Ajoutez le nouveau fichier manifest IA au service d'installation IA, en spécifiant des critères définissant les clients qui doivent utiliser ces instructions d'installation. Reportez-vous à la section “Ajout d'un fichier manifest IA ” à la page 26 pour des exemples d'ajout de fichier manifest IA à un service d'installation. Reportez-vous à la section [Chapitre 3, “Personnalisation des installations”](#) pour plus d'informations sur la spécification de critères de clients.

---

**Remarque** – Si un fichier manifest non valide est fourni à un client, l'installation automatisée est interrompue. Pour rechercher la cause de l'échec de la validation, voir `/tmp/install_log` sur le client.

---

Les balises `<auto_install>` et `<ai_instance>` entourent l'ensemble du fichier manifest IA.

La balise `<ai_instance>` possède les attributs suivants :

- `name` – requis. Vous devez attribuer un nom au fichier manifest. La valeur de nom doit être unique parmi les fichiers manifest IA dans un service d'installation donné.

```
<auto_install>
  <ai_instance name="AI_manifest_name">
    AI_manifest_contents
  </ai_instance>
</auto_install>
```

- `auto_reboot` – facultatif. L'exclusion de l'attribut `auto_reboot` équivaut à la définition de la valeur de l'attribut sur `false`. Par défaut, le programme d'installation automatisée ne réinitialise pas automatiquement le client après l'installation. Pour demander la réinitialisation automatique du client après une installation réussie, spécifiez `auto_reboot="true"`.

---

**Astuce** – La définition d'`auto_reboot` sur `true` n'est pas recommandée dans le cas des clients x86, car l'ordre d'initialisation n'est pas garanti pour les machines x86.

---

```
<auto_install>
  <ai_instance name="AI_manifest_name" auto_reboot="true">
```

```

    AI_manifest_contents
  </ai_instance>
</auto_install>

```

- `http_proxy` – facultatif. L'attribut `http_proxy` permet d'identifier la source du logiciel à installer si le DNS n'est pas utilisé. La valeur de l'attribut `http_proxy` est un URI. Reportez-vous à la section [“Spécification d'un proxy HTTP pour accéder à un référentiel IPS”](#) à la page 61.

Un fichier manifest IA possède en général les sections suivantes dans les balises `<auto_install>` et `<ai_instance>` :

- `<target>` – facultatif. Reportez-vous à la section [“Définition d'une cible pour l'installation”](#) à la page 45.
- `<software>` – requis. Reportez-vous à la section [“Installation de logiciels”](#) à la page 59.
- `<add_drivers>` – facultatif. Reportez-vous à la section [“Identification et installation de pilotes manquants sur une cible d'installation”](#) à la page 63.

## Définition d'une cible pour l'installation

Si vous ne spécifiez pas un emplacement cible sur un client pour l'installation du SE Oracle Solaris, le programme d'installation automatisée sélectionne une cible par défaut.

L'emplacement cible par défaut pour l'installation est le premier disque détecté sur chaque client qui répond aux exigences de taille. Reportez-vous à la section [“Configuration système requise pour les clients”](#) à la page 90. Si la taille d'un disque est supérieure ou égale à la taille recommandée, le programme d'installation sélectionne ce disque en tant que cible d'installation. Si la taille du disque est inférieure à la taille recommandée, le programme d'installation vérifie le disque suivant. Si aucun disque répondant aux exigences de taille n'est détecté, l'installation automatisée échoue pour ce client. Le journal d'installation sous `/tmp/install_log` contient des informations sur le processus de sélection des disques.

Si vous souhaitez spécifier l'emplacement cible sur un client pour l'installation du SE Oracle Solaris, suivez les instructions de cette section.

Utilisez la balise `<target>` pour définir une cible pour l'installation sur chaque client. La balise `<target>` contient la balise `<target_device>`. La balise `<target_device>` comporte les éléments suivants :

- `<disk>` – facultatif. Utilisez cet élément pour définir un disque cible, la partition ou la tranche pour l'installation de tous les clients qui utilisent ce fichier manifest IA. Chaque fichier manifest IA peut spécifier une seule section `<target><target_device><disk>`. Si vous spécifiez une partition ou une tranche, cette spécification doit se produire à l'intérieur d'une spécification de disque. Un disque peut être spécifié par un nom, un ensemble de propriétés pour décrire le disque, un mot-clé ou un périphérique iSCSI. Le disque spécifié est le périphérique root.

Une tranche au sein d'un disque peut être désignée comme étant la tranche root. Si aucune tranche n'est spécifiée, la tranche root est 0, et le programme d'installation automatisée installe le SE Oracle Solaris sur la tranche 0.

Pour un système x86, si le disque est partitionné, la cible d'installation doit être une partition Solaris. Un disque ne peut posséder qu'une seule partition Solaris. Vous pouvez utiliser une partition Solaris existante ou créer une partition Solaris.

Le programme d'installation automatisée utilise les méthodes déterministes et non déterministes pour spécifier un disque cible d'installation.

- “[Spécifications déterministes de disque cible](#)” à la page 47 : la spécification déterministe inclut, par exemple, le nom de disque ou l'ID du périphérique. Spécifiez le mot-clé `boot_disk` afin d'effectuer l'installation sur le disque d'initialisation. Si vous spécifiez la cible à l'aide de spécificateurs déterministes, ou en utilisant le mot-clé `boot_disk`, vous ne pourrez pas utiliser un autre indicateur cible, car la cible est déjà déterminée.
- “[Spécifications non déterministes de disque cible](#)” à la page 50 : la spécification non déterministe inclut, par exemple, le type de périphérique, le fournisseur de périphérique ou la taille de disque. Vous pouvez utiliser plusieurs spécificateurs non déterministes dans un fichier manifest IA.

Vous pouvez spécifier une partition ou une tranche d'un disque de sorte qu'elle soit la cible de l'installation automatisée. Vous pouvez également utiliser le programme d'installation automatisée pour configurer les partitions de disque et les tranches qui ne sont pas la cible de l'installation automatisée. Reportez-vous aux sections “[Configuration du partitionnement sur un client x86](#)” à la page 51 et “[Configuration de tranches sur un disque](#)” à la page 55.

- `<swap>` et `<dump>` – Facultatif. Vous pouvez utiliser le programme d'installation automatisée pour configurer le swap et le vidage pendant l'installation automatisée. Reportez-vous à la section “[Configuration du swap et du vidage sur le périphérique d'installation](#)” à la page 57.

## Spécifications déterministes de disque cible

Chacune des spécifications décrites dans cette section détermine la cible donnée sur laquelle installer le système. Si vous utilisez une spécification décrite dans cette section, vous ne pourrez pas utiliser d'autre spécification cible.

Les éléments de balise dans cette section vous permettent de spécifier un nom de périphérique, un nom de volume, un ID de périphérique, un chemin d'accès de périphérique, un périphérique iSCSI ou le disque d'initialisation.

---

**Remarque** – Les noms de périphériques logiques (`c#t#d#`) peuvent changer selon les versions du système d'exploitation.

---

### EXEMPLE 4-2 Spécification d'un nom de périphérique cible

Utilisez un `name_type` de forme `ctd` pour spécifier un nom de périphérique. Il s'agit de la valeur par défaut si `name_type` n'est pas spécifié.

L'exemple suivant spécifie un nom de périphérique logique.

```
<target>
  <target_device>
    <disk>
      <disk_name name="c1t0d0" name_type="ctd"/>
    </disk>
  </target_device>
</target>
```

L'exemple suivant spécifie un nom MPXIO.

```
<target>
  <target_device>
    <disk>
      <disk_name name="c0t2000002037CD9F72d0" name_type="ctd"/>
    </disk>
  </target_device>
</target>
```

### EXEMPLE 4-3 Spécification d'un nom de volume cible

Utilisez un `name_type` de forme `vol id` pour spécifier un nom de volume.

```
<target>
  <target_device>
    <disk>
      <disk_name name="ai-disk" name_type="vol id"/>
    </disk>
  </target_device>
</target>
```

Le nom de volume peut avoir été défini à l'aide de la commande `format(1M)`, comme indiqué dans l'exemple suivant.

**EXEMPLE 4-3** Spécification d'un nom de volume cible (Suite)

```
$ format -d c0d0 > /dev/null 2>/dev/null - <<EOF
volname
"ai-disk"
y
quit
EOF
```

**EXEMPLE 4-4** Spécification d'un ID de périphérique cible

Utilisez un `name_type` de forme `devid` pour spécifier un ID de périphérique.

```
<target>
  <target_device>
    <disk>
      <disk_name name="id1,sd@n500000e012596560" name_type="devid"/>
    </disk>
  </target_device>
</target>
```

Pour obtenir l'ID du périphérique associé à un nom de périphérique de forme `c#t#d#`, vous pouvez utiliser la commande `iostat(1M)` avec les options `-ien`, par exemple.

```
$ iostat -iEn
c7t0d0          Soft Errors: 0 Hard Errors: 0 Transport Errors: 0
Model: ST31000340NS   Revision: Device Id:
id1,sd@n500000e012596560
...
```

La commande `iostat` ne signale pas l'ID de périphérique sur les invités LDOM. Pour les invités LDOM, utilisez le pilote test de détection de cible comme indiqué dans l'exemple suivant. Ce pilote test est disponible dans le package `IPS system/install/tests` ainsi que sur l'image IA. Initialisez l'image IA pour exécuter cette commande sur un client qui n'a pas accès aux packages IPS.

```
# /opt/install-test/bin/tdmgtst -dv | grep ddm_disk_dev_id
ddm_disk_dev_id=id1,vdc@f8498536e4a8ad037000bcb400001
```

La spécification d'ID de périphérique cible ne fonctionne pas dans xVM/PV, car l'ID de périphérique n'est pas disponible pour les lecteurs virtuels.

**EXEMPLE 4-5** Spécification d'un chemin d'accès au périphérique cible

Utilisez un `name_type` de forme `devpath` pour spécifier un chemin de périphérique sous `/devices`.

```
<target>
  <target_device>
    <disk>
      <disk_name name="/pci@0/pci@9/pci@0/scsi@1/sd@0,0" name_type="devpath"/>
    </disk>
  </target_device>
</target>
```



**EXEMPLE 4-5** Spécification d'un chemin d'accès au périphérique cible (Suite)

```
</target_device>
</target>
```

Le chemin d'accès physique est le chemin d'accès auquel est lié le lien symbolique `c#t#d#s#` sous `devices`, comme indiqué dans l'exemple suivant.

```
$ ls -l /dev/dsk/c2t0d0s0
/dev/dsk/c2t0d0s0 -> ../../devices/pci@7c0/pci@0/pci@1/pci@0/ide@8/sd@0,0:a
```

**EXEMPLE 4-6** Spécification d'une cible iSCSI

La technologie iSCSI permet d'effectuer les installations sur un lecteur de disque hébergé par un autre ordinateur sur le même réseau TCP/IP.

Pour spécifier une cible iSCSI en tant que cible d'installation, vous devez d'abord créer une cible d'initialisation iSCSI. Utilisez la commande `iscsiadm(1M)` pour créer une cible d'initialisation iSCSI et pour obtenir les valeurs de nom et IP pour la balise `iscsi` dans le fichier manifest IA.

Pour spécifier une cible iSCSI pour les installations automatisées, spécifiez un nom de cible iSCSI et une adresse IP comme indiqué dans l'exemple suivant. Le nom de la cible peut être un nom IQN, EUI ou NAA. La valeur de nom ne doit pas contenir plus de 233 caractères.

```
<target>
  <target_device>
    <disk>
      <iscsi name="c0d2E0001010F68">
        <ip>192.168.1.34</ip>
      </iscsi>
    </disk>
  </target_device>
</target>
```

**EXEMPLE 4-7** Spécification du disque d'initialisation en tant que cible

La balise `<disk_keyword>` accepte `boot_disk` en tant que mot-clé. Ce mot-clé spécifie le disque d'initialisation en cours en tant que cible d'installation.

```
<target>
  <target_device>
    <disk>
      <disk_keyword key="boot_disk"/>
    </disk>
  </target_device>
</target>
```

Notez les restrictions suivantes concernant le mot-clé `boot_disk`.

- Le mot-clé `boot_disk` ne fonctionne pas dans xVM/PV, car les informations sur le disque d'initialisation ne sont pas disponibles dans cet environnement.
- Sur certains systèmes x86, le BIOS ne signale pas le disque d'initialisation correctement.

**EXEMPLE 4-7** Spécification du disque d'initialisation en tant que cible (Suite)

- Pour des clients SPARC, si la propriété OBP `diag-switch?` est définie sur `true`, les informations sur le disque d'initialisation ne sont pas disponibles. Pour utiliser le mot-clé `boot_disk` pour les clients SPARC, assurez-vous que `diag-switch?` est défini sur `false`.

```
# eeprom diag-switch?=false
```

## Spécifications non déterministes de disque cible

Vous pouvez utiliser plusieurs spécificateurs non déterministes dans un fichier manifest IA, comme indiqué dans l'exemple suivant.

```
<target>
  <target_device>
    <disk>
      <disk_prop dev_vendor="hitachi" dev_size="20gb"/>
    </disk>
  </target_device>
</target>
```

Utilisez la balise `<disk_prop>` pour spécifier un type de cible, un nom de fournisseur ou une taille cible.

- `dev_type` - Le type du disque cible. Les valeurs possibles sont les suivantes : SCSI, ATA et USB. Cette valeur n'est pas sensible à la casse.
- `dev_vendor` - Le fabricant de l'unité de disque. Les valeurs possibles sont : Sun, Hitachi, EMC, Seagate, Fujitsu. Cette valeur n'est pas sensible à la casse.
- `dev_size` - La taille du périphérique cible. Cette taille doit avoir un suffixe d'unités.

Les suffixes d'unités valides sont les suivants :

- sectors, secs, sec, s, SECTORS, SECS, SEC ou S (secteurs, secs, sec, s, SECTEURS, SECS, SEC ou S)
- MB, megabytes, megabyte, mb, m, MEGABYTES, MEGABYTE, M (Mo, méga-octets, méga-octet, mo, m, MÉGA-OCTETS, MÉGA-OCTET, M)
- GB, gigabytes, gigabyte, gb, g, GIGABYTES, GIGABYTE, G (Go, giga-octets, giga-octet, go, g, GIGA-OCTETS, GIGA-OCTET, G)
- TB, terabytes, terabyte, tb, t, TERABYTES, TERABYTE, T (To, téra-octets, téra-octet, to, t, TÉRA-OCTETS, TÉRA-OCTET, T)

Les exemples suivants présentent deux méthodes différentes pour déterminer le fabricant d'un disque. Notez que ces recherches ne peuvent pas identifier le fabricant des disques ATA.

Utilisez la commande `iostat(1M)` :

```
# iostat -En
c2t1d0      Soft Errors: 0 Hard Errors: 0 Transport Errors: 0
Vendor: HITACHI Product: HUS10733ASUN72G Revision: PA05 Serial No: 0602RW159S
```

Utilisez le pilote test de détection de cible qui est disponible dans l'image IA :

```
# /opt/install-test/bin/test_td -dv
Disk discovery
Total number of disks: 2
-----
num |  name|  vendor| ctype| mtype| rem| lbl| bsize|#of blocks|size [MB]|
-----
  1 | *c2t0d0| SEAGATE| scsi| FIXED| No| V| 512| 71132959| 34732|
  2 | c2t1d0| HITACHI| scsi| FIXED| No| V| 512| 143374738| 70007|
-----
```

## Configuration du partitionnement sur un client x86

Vous pouvez utiliser le fichier manifest IA pour configurer des partitions sur un disque pendant l'installation automatisée. Vous pouvez spécifier une partition en tant que cible d'installation et vous pouvez supprimer des partitions existantes, modifier la taille ou le type d'une partition existante, et créer des partitions.

---

**Remarque** – Les opérations de partition sont utiles pour les clients x86 uniquement.

---

Utilisez la balise <partition> pour configurer le partitionnement d'un disque. La balise <partition> doit être dans une balise <disk>. La balise <partition> possède les attributs suivants :

- **action** – Facultatif. L'attribut **action** possède les valeurs suivantes :
  - **create** – Créer une nouvelle partition. Il s'agit de la valeur par défaut si **action** n'est pas spécifié.
  - **delete** – Supprimer la partition nommée.
  - **use\_existing** – Utiliser la partition Solaris existante en tant que cible d'installation. Si une partition Solaris existe, elle est la cible d'installation par défaut. Si vous spécifiez **use\_existing** et qu'aucune partition Solaris n'existe, l'installation automatisée échoue pour ce client. Une seule action **use\_existing** peut être spécifiée dans un fichier manifest IA.
- **name** - Requis si l'action est **create** ou **delete**. L'attribut **name** désigne le numéro d'identification unique de partition dans la table de partition. Pour les partitions principales, la valeur de **name** peut être 1, 2, 3 ou 4. Une seule partition principale peut être une partition étendue (type de partition DOSEXT). Pour les partitions logiques, **name** peut être un nombre entier compris entre 5 et 36.

Lorsque la valeur `action` est `create`, le programme d'installation automatisée crée la partition avec la valeur `name` spécifiée pendant l'installation automatisée. Lorsque la valeur `action` est `delete`, le programme d'installation automatisée supprime la partition avec la valeur `name` spécifiée pendant l'installation automatisée.

---

**Remarque** – Les modifications de partitionnement apportées pendant l'installation automatisée se produisent suivant l'ordre dans lequel les actions `delete` et `create` sont répertoriées dans le fichier manifest IA.

---

- `part_type` – Facultatif. L'attribut `part_type` indique le type de partition à créer ou à supprimer. La valeur par défaut est 191 (partition Solaris) si `part_type` n'est pas spécifié. Les valeurs valides de `part_type` sont SOLARIS (191), UNIXOS (99), DOSEXT, DOS16, DOSEXTLBA et FAT32.

Lorsque vous créez une partition, vous pouvez, si vous le souhaitez, spécifier la taille de la nouvelle partition. Si une seule partition est spécifiée et que la taille n'est pas spécifiée, les algorithmes suivants sont utilisés pour calculer la taille de la nouvelle partition. Si plus d'une partition est spécifiée, la taille peut alors être omise pour une seule des spécifications de partition.

- Pour les partitions étendues ou principales, la taille par défaut est l'espace libre restant sur le disque.
- Pour une partition logique d'une partition étendue, la taille par défaut est la plus grande partie de l'espace libre dans la partition étendue pour la partition logique.

La balise `<taille>` possède les attributs suivants.

- L'attribut `start_sector` est le numéro de secteur où la partition doit commencer.
- L'attribut `val` est la taille de la partition. Cette taille doit avoir un suffixe d'unités. Consultez la liste de suffixes d'unités dans la description de la propriété de disque `dev_size` dans [“Spécifications non déterministes de disque cible” à la page 50](#).

## Spécification d'une partition en tant que cible d'installation pour un client x86

Sur un client x86, la cible d'installation peut être une partition de disque. Si le disque est partitionné, la cible d'installation doit être une partition Solaris. Une seule partition Solaris peut exister sur un disque. Vous pouvez utiliser une partition Solaris existante ou en créer une nouvelle.

Si une partition existe et que son type est SOLARIS ou 191, cette partition est la cible d'installation par défaut. Si aucune partition Solaris n'existe et que les instructions d'installation ne créent pas de partition Solaris, le programme d'installation automatisée crée une partition Solaris avec l'espace disque restant. Si cet espace est inférieur à l'espace requis pour une installation IA, l'installation échoue pour ce client.

**EXEMPLE 4-8** Spécification d'une partition Solaris existante en tant que cible d'installation

Dans cet exemple, la cible d'installation est une partition Solaris existante. La cible d'installation est le premier disque trouvé qui contient une partition Solaris (type de partition SOLARIS ou 191). Si aucune partition Solaris existante n'est détectée sur un client spécifique, l'installation automatisée échoue pour ce client.

```
<target>
  <target_device>
    <disk>
      <partition action="use_existing"/>
    </disk>
  </target_device>
</target>
```

**EXEMPLE 4-9** Spécification d'une nouvelle partition en tant que cible d'installation

Cet exemple ne spécifie aucune action. L'action par défaut est create. Cet exemple crée deux partitions principales.

```
<target>
  <target_device>
    <disk>
      <partition name="1" part_type="191">
        <size start_sector="200" val="20gb"/>
      </partition>
      <partition name="4" part_type="99">
        <size start_sector="2000" val="20gb"/>
      </partition>
    </disk>
  </target_device>
</target>
```

## Modification des partitions sur un client d'installation x86

En plus de définir une partition en tant que cible d'installation, vous pouvez également supprimer des partitions existantes et en créer de nouvelles pendant l'installation automatisée. Vous pouvez créer une partition étendue ou créer des partitions logiques dans une nouvelle partition ou une partition étendue existante.

---

**Remarque** – Les modifications de partitionnement apportées pendant l'installation automatisée se produisent suivant l'ordre dans lequel les actions delete et create sont répertoriées dans le fichier manifest IA.

---

**EXEMPLE 4-10** Suppression d'une partition existante

Vous pouvez supprimer les partitions existantes lors de l'installation cliente. Spécifiez l'action delete dans la balise <partition> et indiquez le numéro de la partition à supprimer dans l'attribut name.

**EXEMPLE 4-10** Suppression d'une partition existante (Suite)

```
<target>
  <target_device>
    <disk>
      <partition action="delete" name="3"/>
    </disk>
  </target_device>
</target>
```

**EXEMPLE 4-11** Création d'une partition étendue

Vous pouvez créer une partition étendue dans une table de partition `fdisk` en tant que partie de votre installation automatisée. Une partition étendue est une partition principale (1, 2, 3 ou 4) de type `DOSEXT`. Une seule partition principale peut être une partition étendue. Une partition étendue fournit de l'espace pour une ou plusieurs partitions logiques. Plusieurs partitions logiques peuvent être créées dans une partition étendue.

Cet exemple crée une partition étendue de taille maximale.

```
<target>
  <target_device>
    <disk>
      <partition name="3" part_type="DOSEXT"/>
    </disk>
  </target_device>
</target>
```

**EXEMPLE 4-12** Création d'une partition logique

Cet exemple crée une partition logique de type `SOLARIS` au sein de la partition étendue existante, en utilisant l'ensemble de l'espace libre disponible de la partition étendue.

En l'absence d'autres partitions logiques, l'intégralité de la partition étendue est utilisée pour la partition logique. S'il existe plusieurs partitions logiques, la quantité totale d'espace utilisée pour toutes les partitions logiques d'une partition étendue ne doit pas dépasser la quantité totale d'espace de la partition étendue. S'il n'y a pas d'espace disponible pour la partition logique parce qu'une autre partition logique utilise déjà tout l'espace, la création de partition échoue.

```
<target>
  <target_device>
    <disk>
      <partition name="7" part_type="SOLARIS"/>
    </disk>
  </target_device>
</target>
```

**EXEMPLE 4-13** Création d'une partition étendue contenant des partitions logiques

Cet exemple crée une partition étendue contenant deux nouvelles partitions logiques. Cet exemple contient les tâches suivantes.

- Création d'une partition étendue à l'aide de la plus grande partie d'espace disponible sur le disque.
- Création d'une partition logique FAT32 dans la partition étendue, à l'aide de la première partie de 10 Go contiguë de l'espace disponible.
- Création d'une partition logique Solaris à l'aide de la plus grande quantité d'espace libre restant.

```
<target>
  <target_device>
    <disk>
      <!-- Create an extended partition in the largest block of free space -->
      <partition name="4" part_type="DOSEXT"/>

      <!-- Create a 10G FAT32 logical partition on the extended partition.
           This partition consumes the 1st 10G of the extended partition. -->
      <partition name="5" part_type="FAT32">
        <size val="10gb"/>
      </partition>

      <!-- Create a Solaris logical partition using the
           largest unused free space in the extended partition.
           In this example, the partition uses the remaining space
           in the extended partition. -->
      <partition name="6" part_type="SOLARIS"/>
    </disk>
  </target_device>
</target>
```

## Configuration de tranches sur un disque

Vous pouvez utiliser le fichier manifest IA pour configurer des tranches sur un disque pendant l'installation automatisée. Vous pouvez spécifier une tranche en tant que cible d'installation. Vous pouvez créer de nouvelles tranches, mais également supprimer ou conserver des tranches existantes.

Les tranches sont similaires aux partitions. Les différences entre les tranches et les partitions sont les suivantes :

- Une tranche peut être désignée en tant que tranche root.
- Les tranches peuvent être écrasées.
- Les tranches n'ont pas d'action `use_existing`.
- Les tranches ont une action `preserve` qui vous permet de spécifier les tranches à conserver au cours de l'installation.

Utilisez la balise `<slice>` pour configurer des tranches sur un disque. La balise `<slice>` doit se produire à l'intérieur d'une balise `<disk>`. La balise `<slice>` possède les attributs suivants :

- `action` – Facultatif. L'attribut `action` possède les valeurs suivantes :
  - `create` – Créer une tranche. Il s'agit de la valeur par défaut si `action` n'est pas spécifié.
  - `delete` – Supprimer la tranche nommée.
  - `preserve` — Conserver les données sur la tranche nommée pendant l'installation.
- `name` — Requis. L'attribut `name` désigne le numéro d'identification unique de la tranche dans la table VTOC. Les valeurs valides de `name` sont 0, 1, 3, 4, 5, 6 et 7. La valeur 2 n'est pas autorisée car la tranche 2 représente le disque entier. Si aucune tranche n'est spécifiée, la tranche `root` est 0, et le programme d'installation automatisée installe le SE Oracle Solaris sur la tranche 0.
- `is_root` – Facultatif. L'attribut `is_root` peut être appliqué à une tranche au plus. La tranche avec l'attribut `is_root` est incluse dans le `pool root`. La valeur par défaut de l'attribut `is_root` est `false`.
- `force` – Facultatif. L'attribut `force` autorise l'écrasement d'une tranche. La création d'une tranche qui existe déjà est une erreur si vous ne définissez pas `force` sur `true`. Si vous créez une tranche qui existe déjà et que vous spécifiez `force="true"`, la tranche qui vient d'être créée écrase la tranche existante. La valeur par défaut de l'attribut `force` est `false`.

Lorsque vous créez une tranche, vous pouvez, si vous le souhaitez, spécifier sa taille. Si une seule tranche est spécifiée et que la taille n'est pas spécifiée, la tranche correspond à la taille entière du disque. Si plus d'une tranche est spécifiée, la taille peut alors être omise pour l'une des spécifications de tranche. Spécifiez la taille de la tranche dans l'attribut `val` de la balise `<size>` de la tranche. Cette taille doit posséder un suffixe d'unités. Consultez la liste de suffixes d'unités dans la description de la propriété de disque `dev_size` dans [“Spécifications non déterministes de disque cible”](#) à la page 50.

## Spécification d'une tranche en tant que cible d'installation

La cible d'installation peut être une tranche d'un disque.

### EXEMPLE 4-14 Spécification d'une nouvelle tranche en tant que cible d'installation

L'exemple suivant crée deux tranches. Le SE Oracle Solaris sera installé sur la tranche 0. La tranche 0 existait déjà et sera remplacée par la nouvelle spécification.

```
<target>
  <target_device>
    <disk>
      <slice name="0" is_root="true" force="true">
        <size val="20gb"/>
      </slice>
      <slice name="4">
        <size val="20gb"/>
      </slice>
```



**EXEMPLE 4-14** Spécification d'une nouvelle tranche en tant que cible d'installation (Suite)

```
</disk>
</target_device>
</target>
```

## Modification de tranches sur le client d'installation

En plus de définir une tranche en tant que cible d'installation, vous pouvez également créer, supprimer et conserver d'autres tranches pendant l'installation automatisée.

**EXEMPLE 4-15** Conservation d'une tranche

Spécifiez l'action `preserve` dans la balise `<slice>` pour indiquer que cette tranche ne doit pas être modifiée au cours de l'installation. L'exemple suivant explique comment conserver la tranche 4.

```
<target>
  <target_device>
    <disk>
      <slice name="0" is_root="true">
        <size val="20gb"/>
      </slice>
      <slice action="preserve" name="4"/>
    </disk>
  </target_device>
</target>
```

**EXEMPLE 4-16** Suppression d'une tranche

Spécifiez l'action `delete` dans la balise `<slice>` pour supprimer cette tranche au cours de l'installation. L'exemple suivant explique comment supprimer la tranche 4.

```
<target>
  <target_device>
    <disk>
      <slice name="0" is_root="true">
        <size val="20gb"/>
      </slice>
      <slice action="delete" name="4"/>
    </disk>
  </target_device>
</target>
```

## Configuration du swap et du vidage sur le périphérique d'installation

Vous pouvez utiliser le fichier manifest IA pour configurer le swap et le vidage au cours de l'installation automatisée.

Utilisez la balise <swap> pour configurer le swap sur le périphérique d'installation. Utilisez la balise <dump> pour configurer le vidage sur le périphérique d'installation.

La balise <swap> possède un attribut `no_swap` facultatif. La valeur de l'attribut `no_swap` peut être `True` ou `False`. La valeur par défaut est `False` (Faux). Si `no_swap="true"` est spécifié, le swap n'est pas configuré sur le périphérique d'installation.

La balise <dump> possède un attribut `no_dump` facultatif. L'attribut `no_dump` fonctionne de la même façon pour la configuration du vidage que l'attribut `no_swap` pour la configuration du swap.

Les balises <swap> et <dump> utilisent la balise `zvol`. La balise `zvol` possède les attributs suivants :

- `action` – Facultatif. L'action par défaut est `create`, pour configurer le nouveau swap ou vidage.
- `name` – Requis.

Pour spécifier la taille du swap ou du vidage, utilisez la balise `size` dans la balise `zvol`. L'attribut `val` de la balise `size` doit avoir un suffixe d'unités. Consultez la liste de suffixes d'unités dans la description de la propriété de disque `dev_size` dans [“Spécifications non déterministes de disque cible” à la page 50](#).

#### EXEMPLE 4-17 Configuration de swap sur le périphérique d'installation

```
<target>
  <target_device>
    <swap>
      <zvol action="create" name="swap">
        <size val="20gb"/>
      </zvol>
    </swap>
  </target_device>
</target>
```

#### EXEMPLE 4-18 Annulation de la configuration de swap sur le périphérique d'installation

```
<target>
  <target_device>
    <swap no_swap="true"/>
  </target_device>
</target>
```

#### EXEMPLE 4-19 Configuration du vidage sur le périphérique d'installation

```
<target>
  <target_device>
    <dump>
      <zvol action="create" name="dump">
        <size val="2gb"/>
      </zvol>
    </dump>
```

---

**EXEMPLE 4-19** Configuration du vidage sur le périphérique d'installation (Suite)

```
</target_device  
</target>
```

## Installation de logiciels

L'image d'installation IA n'est pas une installation complète. Les machines client doivent accéder à un référentiel de packages IPS pour terminer leur installation. Le fichier manifest IA doit indiquer au moins un emplacement du référentiel de packages, et vous devez spécifier les noms des packages à installer.

Utilisez la balise `<software>` pour définir les emplacements de référentiels de packages et les noms des packages à installer.

La balise `<logiciel>` comporte les éléments suivants :

- `<source>` – Requis. Utilisez cet élément pour spécifier où IA doit obtenir les packages à installer. Reportez-vous à la section “[Spécification d'une source de packages à installer](#)” à la page 59.
- `<software_data>` – Requis. Utilisez cet élément pour spécifier les noms des packages à installer. Un ensemble particulier de packages est requis pour chaque installation automatisée. Vous pouvez choisir d'installer des packages supplémentaires. Reportez-vous à la section “[Spécification des packages à installer](#)” à la page 61.

Un fichier manifest IA ne doit pas nécessairement posséder un nombre égal d'éléments `<source>` et `<software_data>`. Un élément `<software>` peut avoir plus d'éléments `<source>` que `<software_data>`, ou un élément `<software>` peut avoir plus d'éléments `<software_data>` que `<source>`.

## Spécification d'une source de packages à installer

Un *référentiel* est un emplacement dans lequel les packages IPS sont publiés et à partir duquel les packages sont extraits. Un référentiel peut se trouver sur le réseau local ou sur Internet. Une personne ou une société qui place un package dans un référentiel est appelé un *éditeur*. Chaque fichier manifest IA doit spécifier au moins un référentiel de packages IPS pour installer le SE Oracle Solaris.

---

**Remarque** – La version SE Oracle Solaris de l'image d'initialisation IA et des packages IPS doit être la même. Le ou les référentiels IPS spécifiés dans le fichier manifest IA doivent contenir des packages pour cette version.

---

Utilisez la balise `<source>` pour spécifier un emplacement où le programme d'installation automatisée doit obtenir les packages à installer. Un élément `<software>` peut contenir plusieurs éléments `<source>`. Chaque élément `<source>` peut spécifier un éditeur de package IPS ou un chemin d'accès d'un répertoire.

- `<publisher>` — Chaque fichier manifest IA doit indiquer au moins un élément `<publisher>`, car un ensemble spécifique de packages IPS est requis pour chaque installation automatisée d'Oracle Solaris. L'élément `<publisher>` possède un attribut `name` facultatif. L'élément `<publisher>` comporte les éléments suivants :
  - `<origin>` – Requis. L'élément `<origin>` possède un attribut `name` requis qui donne l'URI principal de ce référentiel de packages IPS. Un élément `<publisher>` ne peut avoir qu'un seul élément `<origin>`.
  - `<mirror>` – Facultatif. L'élément `<mirror>` possède un attribut `name` requis qui donne un URI supplémentaire pour ce même référentiel de packages IPS. Un élément `<publisher>` peut posséder plus d'un élément `<mirror>`.

Un miroir est différent de plusieurs référentiels. Un miroir est un emplacement distinct pour un référentiel qui possède le même contenu que le référentiel `<origin>` dans le même élément `<publisher>`. Pour spécifier d'autres référentiels avec du contenu différent, utilisez d'autres éléments `<source>`.

---

**Remarque** – Le premier référentiel IPS spécifié dans un fichier manifest IA est la source préférée pour tous les packages IPS. Si un package IPS spécifique n'est pas trouvé dans le référentiel préféré, il est recherché dans d'autres référentiels IPS suivant l'ordre dans lequel ils sont spécifiés dans le fichier manifest IA. Voir aussi [Exemple 4–23](#).

---

- `<dir>` – Facultatif. Chaque élément `<source>` doit être posséder un élément `<publisher>` ou un élément `<dir>`. L'élément `<dir>` possède un attribut obligatoire `path` qui donne le chemin d'accès à un ou plusieurs packages à installer.

#### EXEMPLE 4–20 Spécification de référentiels de package IPS

Dans cet exemple, `solaris` est l'éditeur de package IPS préféré. Il s'agit du référentiel qui sera recherché en premier pour les packages IPS. Ce référentiel doit contenir les packages requis pour installer le SE Oracle Solaris. Ce référentiel doit contenir des packages pour le SE Oracle Solaris dont la version doit être la même que l'image d'initialisation IA associée à ce service d'installation.

Le référentiel `mirror_repo` contient exactement le même contenu que le référentiel `solaris`. Ce référentiel sera utilisé si la connexion au référentiel `solaris` est trop lente.

Le contenu du référentiel `additional_repo` est différent de celui du référentiel `solaris`. Ce référentiel sera utilisé si un package IPS spécifié n'est pas détecté dans le référentiel `solaris`. Le référentiel `additional_repo` peut être un référentiel sur le réseau local qui contient les packages IPS personnalisés, par exemple.

**EXEMPLE 4-20** Spécification de référentiels de package IPS (Suite)

```

<software name="IPS">
  <source>
    <publisher name="solaris">
      <origin name="http://pkg.oracle.com/solaris/release"/>
      <mirror name="http://pkg.mirror_repo"/>
    </publisher>
  </source>
  <source>
    <publisher>
      <origin name="http://pkg.additional_repo"/>
    </publisher>
  </source>
</software>

```

**Spécification d'un proxy HTTP pour accéder à un référentiel IPS**

Un client doit accéder à un référentiel IPS pour installer le SE Oracle Solaris. Dans la configuration recommandée décrite dans [Chapitre 1, “Présentation du programme d'installation automatisée”](#), un serveur DHCP envoie les informations DNS au client. Ces informations DNS sont utilisées pour résoudre l'URI du référentiel IPS en adresse IP.

Si un client n'a pas de connexion directe au serveur du référentiel IPS, le fichier manifest IA peut spécifier un serveur proxy pour permettre au client de sortir de son réseau. Ce proxy s'applique à tous les éditeurs.

La valeur de l'attribut `http_proxy` de l'élément `<ai_instance>` est l'URL du proxy comme suit :

```

<auto_install>
  <ai_instance name="AI_manifest_name" http_proxy="http://192.168.0.101:8080">
    AI_manifest_contents
  </ai_instance>
</auto_install>

```

**Spécification des packages à installer**

L'image d'initialisation IA n'est pas une installation complète. Les machines client doivent accéder à un référentiel de packages IPS pour terminer leur installation. Le fichier manifest IA par défaut indique le minimum de packages à installer pour l'installation du SE Oracle Solaris. Affichez la liste des packages à installer dans un élément `<software_data>` dans un élément `<software>` l'élément dans le fichier manifest IA.

La balise `<software_data>` possède les attributs suivants :

- `action` – Facultatif. La valeur `install` est la valeur par défaut si `action` n'est pas spécifié. Les autres actions possibles sont `uninstall` et `noinstall`. L'[Exemple 4–21](#) aborde l'utilisation de `uninstall`. "[Identification et installation de pilotes manquants sur une cible d'installation](#)" à la page 63 traite de l'utilisation de `noinstall`.
- `type` – Facultatif. La valeur `IPS` est la valeur par défaut si `type` n'est pas spécifié. `IPS` est le seul type valide de package dans cette section du fichier manifest IA. "[Identification et installation de pilotes manquants sur une cible d'installation](#)" à la page 63 présente des exemples d'installation de différents types de packages dans la section `<add_drivers>` du fichier manifest IA.

L'élément `<software_data>` contient des éléments `<name>`. Chaque élément `<name>` spécifie le nom d'un package à installer. Pour les packages IPS, l'élément `<name>` peut spécifier le référentiel à utiliser. Reportez-vous à l'exemple [Exemple 4–23](#).

#### EXEMPLE 4–21 Spécification du minimum de packages IPS à installer

Cet exemple illustre le minimum de packages IPS à installer pour l'installation du SE Oracle Solaris.

Cet exemple spécifie également les packages à désinstaller. Les packages à désinstaller sont des packages de groupe. Les packages de groupe sont des définitions de package qui sont utilisées pour des raisons de commodité pour installer un ensemble d'autres packages. Vous ne pouvez pas désinstaller un package qui a été installé à l'aide d'un package de groupe, sauf si vous désinstallez préalablement la définition de package de groupe. La désinstallation de la définition de package de groupe ne désinstalle pas tous les packages installés avec le package de groupe. Le package `babel_install` est un package de groupe qui installe `slim_install`, qui est un autre package de groupe. Après l'installation, `slim_install` et `babel_install` sont désinstallés afin qu'un utilisateur puisse ultérieurement désinstaller d'autres packages qui ont été installés en tant qu'éléments de `slim_install` et `babel_install`. La possibilité de désinstaller les packages est nécessaire pour la mise à jour et la personnalisation de l'installation. La définition du package `babel_install` doit être désinstallée en premier afin que la définition du package `slim_install` puisse être désinstallée.

```
<software name="IPS">
  <software_data type="IPS">
    <name>pkg:/entire</name>
    <name>pkg:/babel_install</name>
  </software_data>
  <software_data action="uninstall" type="IPS">
    <name>pkg:/babel_install</name>
    <name>pkg:/slim_install</name>
  </software_data>
</software>
```

**EXEMPLE 4-22** Spécification de packages IPS supplémentaires à installer

Dans cet exemple, Oracle Solaris Studio et le plug-in d'interface graphique NetBeans DTrace sont installés.

```
<software name="IPS">
  <software_data type="IPS">
    <name>pkg:/developer/sunstudio12u1</name>
    <name>pkg:/developer/netbeans/plugin/nb-dtrace</name>
  </software_data>
</software>
```

**EXEMPLE 4-23** Spécification du référentiel IPS dans le nom du package

Dans cet exemple, la suite d'outils OpenOffice et l'utilitaire de messagerie électronique et de gestion d'agenda Evolution sont installés à partir du référentiel IPS `example.com`.

```
<software name="IPS">
  <source>
    <publisher name="solaris">
      <origin name="http://pkg.oracle.com/solaris/release"/>
    </publisher>
  </source>
  <source>
    <publisher name="example.com">
      <origin name="http://pkg.example.com/release"/>
    </publisher>
  </source>
  <software_data type="IPS">
    <name>pkg://example.com/openoffice</name>
    <name>pkg://example.com/mail/evolution</name>
  </software_data>
</software>
```

## Identification et installation de pilotes manquants sur une cible d'installation

L'installation par défaut peut ne pas inclure les pilotes pour tous les périphériques d'un client. Dans le fichier manifest IA, vous pouvez demander au programme d'installation automatisée d'effectuer l'une des tâches suivantes, ou les deux :

- Analyse d'un client pour détecter les pilotes de périphérique manquants et installation des packages de pilotes adéquats pour ce client.
- Installation de packages de pilote nommés de façon explicite sur chaque client qui utilise le fichier manifest IA.

Utilisez la balise `<add_drivers>` afin qu'elle indique au programme d'installation automatisée d'effectuer l'une de ces tâches. Dans les deux cas, les pilotes sont, en premier lieu, ajoutés à l'environnement d'initialisation et installés ultérieurement sur la cible lors du processus

d'installation automatisée. Le fait de commencer par l'ajout de pilotes manquants à l'environnement d'initialisation permet au programme d'installation automatisée d'accéder à tous les périphériques client au cours de l'installation.

---

**Remarque** – Les packages spécifiés dans la section `<add_drivers>` peuvent installer uniquement les fichiers du noyau. Certains systèmes de fichiers sur l'image live sont montés en lecture seule. L'installation sur des systèmes de fichiers en lecture seule, comme `/usr`, échouera. Sur les images live Oracle Solaris, l'ensemble du fichier `/kernel` est en lecture-écriture. La plupart des pilotes sont installés dans `/kernel`.

---

Un fichier manifest IA ne peut pas spécifier plus d'un élément `<add_drivers>`. L'élément `<add_drivers>` peut comporter l'un ou les deux éléments suivants :

- `<software>` – Facultatif. Utilisez cet élément pour spécifier les sources et les noms pour les packages qui sont nécessaires pour effectuer l'installation. Tous les éléments `<software>` spécifiés dans un élément `<add_drivers>` doivent être spécifiés avant l'élément `<search_all>`.

Cet élément `<software>` comprend un élément `<source>` et un élément `<software_data>`. La spécification de l'élément `<source>` est affectée par la valeur du type `<software_data>`.

L'élément `<software_data>` possède un attribut de package `type` et un attribut `action`.

- `type` – Requis. Les valeurs possibles sont `P5I`, `SVR4` et `DU`. Voir [Exemple 4–25](#), [Exemple 4–26](#) et [Exemple 4–27](#).
- `action` – Facultatif. La valeur `install` est la valeur par défaut si `action` n'est pas spécifiée. La seule autre action possible est `noinstall`. Si vous avez défini l'attribut `action` sur `noinstall`, les packages ne sont pas installés sur le périphérique cible. Les packages sont toujours ajoutés à l'environnement d'initialisation et utilisés pour effectuer l'installation.
- `<search_all>` – Facultatif. Un fichier manifest IA ne peut pas spécifier plus d'un élément `<search_all>`. Si un élément `<search_all>` est spécifié, il doit être spécifié après tous les éléments `<software>` qui énumèrent des packages de pilotes explicites.

L'élément `<search_all>` indique au programme d'installation de rechercher dans le client les périphériques qui n'ont actuellement pas de pilote dans l'image d'installation initialisée. Tous les pilotes qui sont détectés, et qui ne sont pas identifiés comme des pilotes tiers sont installés dans l'environnement d'initialisation, puis installés ultérieurement sur le périphérique cible pour ce client.

Un pilote *tiers* dans le programme d'installation automatisée est un pilote qui n'est pas trouvé dans le référentiel mais dont on sait qu'il existe ailleurs, par exemple, sur un site Web.

Si l'attribut facultatif `addall` est défini sur `true` (voir l'[Exemple 4–24](#)), tout pilote tiers nécessaire détecté est également ajouté à l'environnement d'initialisation pour être installé sur la cible. La valeur par défaut de l'attribut `addall` est `false`. Si l'attribut `addall` n'est pas spécifié ou s'il est défini sur `false`, les pilotes tiers manquants ne sont pas installés.



L'élément `<search_all>` possède un élément `<source>` facultatif où vous pouvez spécifier un référentiel de packages dans lequel rechercher des pilotes manquants. Un élément `<search_all>` ne peut pas comporter plus d'un élément `<source>`. Seul un éditeur peut être spécifié au sein d'un élément `<search_all>`. L'élément `<source>` au sein d'un élément `<search_all>` ne peut pas avoir d'élément `<mirror>`.

Si l'élément `<search_all>` ne spécifie pas d'élément `<source>`, les sources spécifiées ailleurs dans ce fichier manifest IA sont alors utilisées. La base de données de chaque référentiel identifie un pilote en tant que pilote tiers ou non.

Si un pilote manquant n'est pas détecté au cours d'une recherche, un message d'avertissement s'affiche, mais l'installation est lancée. Si l'installation est effectuée avec succès, le système redémarre si le fichier manifest IA demande une réinitialisation (`auto_reboot="true"`).

Si un package de pilote indiqué de façon explicite n'est pas trouvé, ou qu'un problème se produit au cours de l'installation de n'importe quel package de pilote (qu'il ait été détecté suite à une recherche ou explicitement spécifié), cette installation de client spécifique est terminée, puis un message s'affiche à propos du pilote manquant et attend la réponse à une invite. Le système ne se réinitialise pas automatiquement lorsque l'installation est terminée. Ce comportement offre la possibilité d'évaluer le message et de déterminer si la réinitialisation réussira. Si une réinitialisation peut réussir sans le pilote manquant, vous pouvez effectuer une réinitialisation manuelle.

#### EXEMPLE 4-24 Recherche de pilotes manquants

L'élément `<search_all>` indique au programme d'installation automatisée de rechercher les périphériques auxquels il manque des pilotes et de rechercher les pilotes nécessaires dans une base de données. Cet exemple spécifie un référentiel dans lequel rechercher les pilotes nécessaires. Cet exemple spécifie `addall="true"` afin d'installer des pilotes qui sont identifiés dans la base de données en tant que pilotes tiers, ainsi que les pilotes qui ne sont pas identifiés en tant que pilotes tiers.

```
<add_drivers>
  <search_all addall="true">
    <source>
      <publisher name="example.com">
        <origin name="http://pkg.example.com/release"/>
      </publisher>
    </source>
  </search_all>
</add_drivers>
```

**EXEMPLE 4-25** Ajout d'un package à l'aide d'un fichier P5I

Cet exemple ajoute un package pkg(5) explicitement spécifié par le biais d'un fichier P5.

Un type de données logicielles P5I est doté des caractéristiques suivantes :

- L'élément <software\_data> ne peut pas avoir un élément <name>.
- Le nom d'origine de l'éditeur contient le chemin d'accès complet au fichier P5I. Le fichier P5I contient le nom du package et d'un référentiel. Le chemin d'accès peut être dans un fichier local ou à un emplacement HTTP ou FTP.

```
<add_drivers>
  <software>
    <source>
      <publisher>
        <origin name="http://myserver.example.com/drivers/p5i/0/mydriver.p5i"/>
      </publisher>
    </source>
    <software_data type="P5I"/>
  </software>
</add_drivers>
```

**EXEMPLE 4-26** Ajout d'un package SVR4

Cet exemple ajoute un package SVR4 spécifié explicitement. Le chemin complet name dans la balise <origin> dans l'élément <source> est le répertoire qui contient les packages. La balise <name> dans l'élément <software\_data> est le nom du package. Pour les packages locaux, ce <name> peut être le répertoire de niveau supérieur d'une arborescence de fichiers du package, ou <name> peut être un fichier de flux de données. Pour les packages distants, <name> est un fichier de flux de données.

Cet exemple illustre l'action noinstall, bien que l'action noinstall puisse être spécifiée dans n'importe quelle balise software\_data . L'action noinstall installe le package uniquement dans l'environnement initialisé. Le package est disponible au cours de l'installation mais n'est pas installé sur le périphérique cible.

```
<add_drivers>
  <software>
    <source>
      <publisher>
        <origin name="/export/package_dir"/>
      </publisher>
    </source>
    <software_data type="SVR4" action="noinstall">
      <name>mydriver2.d</name>
    </software_data>
  </software>
</add_drivers>
```

**EXEMPLE 4-27** Ajout d'une image

Cet exemple ajoute une image de mise à jour des pilotes (DU) ou de mise à jour d'installation (ITU) spécifiée explicitement. S'il s'agit d'une image locale, le chemin name dans la balise

**EXEMPLE 4-27** Ajout d'une image (Suite)

<origin> dans l'élément <source> est le parent du répertoire DU de l'image d'installation si l'image est développée. Si l'image d'installation n'est pas étendue, ce chemin name est le nom de l'image .iso. Si l'image est distante, ce chemin d'accès name est le chemin HTTP ou FTP complet d'un fichier d'image .iso. Tous les packages dans l'image seront ajoutés.

Lorsque le type de données logicielles est DU, l'élément <software\_data> ne peut pas posséder un élément <name>.

```
<add_drivers>
  <software>
    <source>
      <publisher>
        <origin name="/export/duimages/mydriverDU"/>
      </publisher>
    </source>
    <software_data type="DU"/>
  </software>
</add_drivers>
```

## Fichier manifest IA annoté

Le fichier suivant montre un fichier manifest IA qui illustre la plupart des balises et contient des commentaires sur l'utilisation.

```
<?xml version="1.0"?>
<!--
Copyright (c) 2008, 2010, Oracle and/or its affiliates. All rights reserved.
-->
<!--
=====
DTD sample manifest for Automatic Installer input manifest specification.
=====
-->
<!DOCTYPE auto_install SYSTEM "file:///usr/share/auto_install/ai.dtd">
<auto_install>
  <!--
    "auto_reboot" set to "true" may be an issue for x86 machines.
    The boot order is not guaranteed and may cause unexpected
    behavior. If auto_reboot is not desired, removing this
    attribute, e.g. <ai_instance name="sample_ai_manifest">
    will work. "auto_reboot" is set to false by default.
  -->
  <ai_instance name="sample_ai_manifest" auto_reboot="true">
    <!--
      =====
      <target/target_device> - selections for AI target Device specification

      Disk criteria are divided into three mutually exclusive groups:

      G1 - deterministic disk criteria
```

```

.....
* target_device/disk/iscsi parameters
* target_device/disk/disk_name, with name_type attribute:
  one of ctd, volid, devpath or devid

G2 - non-deterministic disk criteria
.....
* target_device/disk/disk_prop: Any of dev_type, dev_vendor or
  dev_size

G3 - keyword disk criteria
.....
* target_device/disk/disk_keyword: "boot_disk"

Schema ai.dtd enforces following policy:

* criteria in group G1 are mutually exclusive - only
  one can be specified at a time

* groups G1, G2 and G3 are mutually exclusive - i.e.
  if criteria from G1 is specified, no criteria
  from G2 or G3 are allowed and vice versa

* multiple criteria from G2 can be specified
=====
-->
<target>
<target_device>
<disk>
  <!-- G1 -->
  <!--
    c##t##d## device name like c0t0d0 or
    MPXIO name like c0t2000002037CD9F72d0
  -->
  <disk_name name="c1t0d0" name_type="ctd"/>
  <!-- volume name set for instance by means
    of format(1M) command
  -->
  <!--
  <disk_name name="ai-disk" name_type="volid"/>
  -->
  <!-- device id - e.g. can be obtained by means of
    iostat(1M) -iEn
  -->
  <!--
  <disk_name name="id1,cmdk@AST31000340NS=_____9QJ2LNYY" name_type="devid"/>
  -->
  <!-- device path under /devices directory, e.g.
    /pci@1e,600000/pci@0/pci@9/pci@0/scsi@1/sd@0,0
  -->
  <!--
  <disk_name name="/pci@0/pci@9/pci@0/scsi@1/sd@0,0" name_type="devpath"/>
  -->
  <!--
    ISCSI target device

  <iscsi name="c0d2E0001010F68">
    <ip>192.168.1.34</ip>
  </iscsi>

```

```

-->
<!-- G2 -->
<!--
<disk_prop dev_vendor="hitachi" dev_size="20480mb"/>
-->
<!-- G3 -->
<!--
<disk_keyword key="boot_disk"/>
-->
<!--
    Uncomment this to force AI to find an existing Solaris
    partition instead of creating a new one.
-->
<!--
<partition action="use_existing"/>
-->
<partition name="1" part_type="99">
  <size start_sector="200" val="20480mb"/>
</partition>
<partition name="4" part_type="99">
  <size start_sector="2000" val="20480mb"/>
</partition>
<slice name="0" is_root="true">
  <size val="20480mb"/>
</slice>
<slice name="4">
  <size val="20480mb"/>
</slice>
</disk>
</target_device>
</target>
<software name="ips">
  <source>
    <publisher name="solaris">
      <origin name="http://pkg.oracle.com/solaris/release"/>
    </publisher>
  </source>
</software_data>
<!--
    By default the latest build available, in the
    specified IPS repository, is installed.
    if another build is required, the build number has
    to be appended to the 'entire' package in following
    form:
<name="entire@0.5.11-0.build#"/>
-->
<software_data type="IPS">
  <name>pkg:/entire</name>
  <name>pkg:/babel_install</name>
</software_data>
<!--
    babel_install and slim_install are group packages used to
    define the default installation. They are removed here so
    that they do not inhibit removal of other packages on the installed
    system
-->
<software_data action="uninstall" type="IPS">
  <name>pkg:/babel_install</name>
  <name>pkg:/slim_install</name>
</software_data>

```

```
</software>
<add_drivers>
  <!--
```

Driver Updates: This section is for adding driver packages to the boot environment before the installation takes place. The installer can then access all devices on the system. The packages installed in the boot environment will also be installed on the target.

A `<search_all>` entry performs a search for devices which are missing their drivers. A repository publisher and location may be specified, and that repository and its database will be used. If no publisher and location is specified, the configured repositories will be used. (See `pkg publisher` command.) If `<addall>` is specified as "true", then drivers the database says are third-party drivers will be added like all others; otherwise third-party drivers will not be added.

```
    <search_all addall="true">
      <source>
        <publisher name="solaris">
          <origin name="http://pkg.oracle.com/solaris/release"/>
        </publisher>
      </source>
    </search_all>
```

`<software>` entries are user-provided specifications of packages needed in order to perform the install. types are P5I, SVR4, DU. A `<software_data>` action of "noinstall" inhibits adding to target.

P5I: A `pkg(5)` P5I file, full path is in the `source/publisher/origin`. Path may be to a local file or an http or ftp specification.

```
    <software>
      <source>
        <publisher>
          <origin
            name=
"http://pkg.oracle.com/solaris/release/p5i/0/driver/firewire.p5i"/>
          </publisher>
        </source>
      <software_data type="P5I"/>
    </software>
```

SVR4: An SVR4 package spec. The `source/publisher/origin` corresponds to the directory containing the packages. The `software/software_data/name` refers to the package's top level directory or the package's datastream file.

```
    <software>
      <source>
        <publisher>
          <origin name="/export/package_dir"/>
        </publisher>
      </source>
      <software_data type="SVR4">
        <name>my_disk_driver.d</name>
      </software_data>
    </software>
```

DU: An ITU (Install Time Update) or Driver Update image.  
The source/publisher/origin refers to the path just above the image's DU directory (if expanded) or the name of the .iso image. All packages in the image will be added.

```
    <software>
      <source>
        <publisher>
          <origin name="/export/duimages/mydriver.iso"/>
        </publisher>
      </source>
      <software_data type="DU"/>
    </software>
  -->
</search_all/>
</add_drivers>
</ai_instance>
</auto_install>
```





# Configuration du système client

---

Ce chapitre décrit la procédure de spécification des informations requises pour configurer le système client après l'installation. Vous pouvez indiquer la configuration de tous les éléments configurables par le biais des propriétés `smf(5)`.

## Création d'un fichier manifest SC personnalisé

Le fichier manifest de configuration système (fichier manifest SC) spécifie la configuration système du client sous la forme d'un ensemble de paramètres de configuration dans le formulaire d'un profil SMF (Service Management Facility, utilitaire de gestion des services). Le fichier manifest SC définit les propriétés SMF pour les services SMF appropriés.

Un profil SMF contenant la configuration système est appliqué au cours de la première initialisation du système après l'installation. Les services SMF responsables de zones de configuration spécifiques traitent les propriétés SMF et configurent le système en conséquence.

- Vous pouvez incorporer un fichier manifest SC dans votre fichier manifest IA. Voir [“Fichier manifest IA par défaut” à la page 37](#) pour obtenir un exemple de fichier manifest SC incorporé dans un fichier manifest IA.

```
<auto_install>
  <ai_instance name="default">
    AI_manifest_content
    <sc_embedded_manifest name="AI">
      <!-- <?xml version='1.0'?>
        SC_manifest_content
      -->
    </sc_embedded_manifest>
  </ai_instance>
</auto_install>
```

- Vous pouvez faire référence à un autre fichier manifest SC dans le fichier manifest IA.

```
<auto_install>
  <ai_instance name="default">
    AI_manifest_content
```

```

    <sc_manifest_file name="AI" URI="./sc_manifest1.xml"/>
  </ai_instance>
</auto_install>

```

## Spécification de la configuration dans un fichier manifest SC

Vous pouvez indiquer la configuration de tous les éléments configurables par le biais des propriétés smf(5). Par exemple, le fichier manifest SC peut configurer un compte root, un utilisateur initial, le clavier, le type de terminal, une interface réseau IPv4 (statique ou DHCP) et une route par défaut, une interface réseau IPv6 (statique ou `addrconf`) et une route par défaut, et un DNS (liste de serveurs de noms, liste de recherche, domaine).

### Comptes root et utilisateur

Le service SMF `svc:/system/install/config` configure les comptes utilisateur et root. Ce service reconnaît deux groupes de propriétés :

- Le groupe de propriétés `root_account` comprend les propriétés SMF qui permettent de configurer le compte root.
- Le groupe de propriétés `user_account` comprend les propriétés SMF qui permettent de configurer les comptes utilisateur.

---

**Astuce** – L'une des méthodes de génération des mots de passe chiffrés pour le SE Oracle Solaris consiste à créer un utilisateur avec le nom et le mot de passe prévus, à copier le mot de passe à partir du fichier `/etc/shadow` entre le 1er et le 2e deux-points de l'enregistrement de l'utilisateur, puis à ajouter ces informations aux valeurs de mot de passe du fichier manifest.

---

### Configuration du compte root

Le groupe de propriétés `root_account` peut contenir les propriétés suivantes.

TABLEAU 5-1 Propriétés du groupe de propriétés `root_account`

Propriétés	Type	Requis	Description
<code>password</code>	<code>aststring</code>	oui	Mot de passe root chiffré.
<code>type</code>	<code>aststring</code>	no	Type de compte : <code>normal</code> ou <code>role</code> . La valeur par défaut est <code>normal</code> .
<code>expire</code>	<code>string</code>	no	Date d'expiration pour la connexion. Si l'option est définie sur 0 (zéro), l'utilisateur sera obligé de modifier le mot de passe root lors de la prochaine connexion.

**EXEMPLE 5-1** Configuration du compte root uniquement après expiration du mot de passe

```
<service name="system/install/config" version="1" type="service">
  <instance name="default" enabled="true">
    <property_group name="root_account" type="application">
      <propval name="password" type="astiring" value="encrypted_password"/>
      <propval name="type" type="astiring" value="normal"/>
      <propval name="expire" type="astiring" value="0"/>
    </property_group>

    <property_group name="other_sc_params" type="application">
      <propval name="timezone" type="astiring" value="GMT"/>
      <propval name="hostname" type="astiring" value="solaris"/>
    </property_group>
  </instance>
</service>
```

## Configuration d'un compte utilisateur

Le groupe de propriétés `user_account` peut contenir les propriétés suivantes.

**TABLEAU 5-2** Propriétés du groupe de propriétés `user_account`

Propriétés	Type	Requis	Description
<code>login</code>	<code>astiring</code>	yes	Identifiant de l'utilisateur.
<code>password</code>	<code>astiring</code>	yes	Mot de passe utilisateur chiffré.
<code>description</code>	<code>astiring</code>	no	En général, le nom complet de l'utilisateur.
<code>shell</code>	<code>astiring</code>	no	Chemin d'accès complet du programme utilisé en tant que shell de l'utilisateur lors de la connexion.
<code>uid</code>	<code>count</code>	no	UID du nouvel utilisateur. L'UID par défaut est 101.
<code>gid</code>	<code>count</code>	no	Appartenance de l'utilisateur au groupe principal. Le GID par défaut est 10.
<code>type</code>	<code>astiring</code>	no	Type de compte : <code>normal</code> ou <code>role</code> . La valeur par défaut est <code>normal</code> .
<code>profiles</code>	<code>astiring</code>	no	Un ou plusieurs profils d'exécution (séparés par des virgules) définis dans <code>prof_attr(4)</code> .
<code>roles</code>	<code>astiring</code>	no	Un ou plusieurs rôles (séparés par des virgules) définis dans <code>user_attr(4)</code> .
<code>sudoers</code>	<code>astiring</code>	no	Entrée indiquée avec l'identifiant dans le fichier <code>sudoers(4)</code> .
<code>expire</code>	<code>astiring</code>	no	Date d'expiration pour la connexion. Si l'option est définie sur 0 (zéro), l'utilisateur est obligé de modifier le mot de passe lors de la prochaine connexion.

TABLEAU 5-2 Propriétés du groupe de propriétés user\_account (Suite)

Propriétés	Type	Requis	Description
home_zfs_dataset	astiring	no	Jeu de données ZFS du répertoire d'accueil de l'utilisateur. La valeur par défaut est <i>root_pool/export/home/identifiant</i>
home_mountpoint	astiring	no	Point de montage du répertoire d'accueil de l'utilisateur. La valeur par défaut est <i>/export/home/identifiant</i>

## Type de terminal et disposition du clavier

Le service SMF `svc:/system/console-login` configure le type de terminal. Voir la page de manuel `ttymon(1M)` pour la définition des propriétés SMF associées.

### EXEMPLE 5-2 Configuration du type de terminal vt100

```
<service name="system/console-login" version="1" type="service">
  <property_group name="ttymon" type="application">
    <propval name="terminal_type" type="astiring" value="vt100"/>
  </property_group>
</service>
```

Le service SMF `svc:/system/keymap` configure la disposition du clavier. Reportez-vous à la page de manuel `kbd(1)` pour la définition des propriétés SMF associées.

### EXEMPLE 5-3 Configuration de la disposition du clavier tchèque

```
<service name='system/keymap' version='1' type='service'>
  <instance name='default' enabled='true'>
    <property_group name='keymap' type='system'>
      <propval name='layout' type='astiring' value='Czech' />
    </property_group>
  </instance>
</service>
```

## Adresse IP statique et DNS

La mise en réseau peut être configurée automatiquement ou manuellement sur un système Oracle Solaris 11 Express. Les configurations que vous pouvez effectuer manuellement peuvent également être spécifiées dans un fichier manifest SC.

La configuration automatique du réseau (NWAM) est activée par le biais de l'activation de `svc:/network/physical:nwam` et de la désactivation de `svc:/network/physical:default`. La structure NWAM gère la configuration du réseau.

La configuration réseau manuelle est activée lorsque `svc:/network/physical:nwam` est désactivé et que `svc:/network/physical:default` est activé. En mode manuel, les paramètres réseau suivants peuvent être configurés :

- Une interface réseau IPv4 , avec une adresse IPv4 statique ou un serveur DHCP configuré
- Route par défaut IPv4
- Une interface réseau IPv6, avec une adresse IPv6 statique ou configurée automatiquement
- Route IPv6 par défaut
- DNS en tant que service d'attribution de noms

Les services SMF `svc:/network/install` et `svc:/network/dns/install` contiennent les propriétés qui leur permettent de configurer une interface réseau physique initiale ou une configuration de client DNS initial. Ces services sont initialement désactivés avec des valeurs de propriété qui n'entraînent aucune configuration du système. Ces services peuvent être activés, et les propriétés appropriées peuvent être configurées dans le fichier manifest SC.

Le service `svc:/network/install` prend en charge la configuration d'une interface IPv4 et d'une interface IPv6 et, le cas échéant, une route par défaut accessible par ces interfaces. Le service définit deux groupes de propriétés : un groupe de propriétés d'une interface IPv4 et un autre pour une interface IPv6. Le service utilise ses propriétés et `ipadm(1M)` pour configurer les interfaces réseau. De même, le service utilise ses propriétés et `route(1M)` pour définir une route par défaut.

Le groupe de propriétés `install_ipv4_interface` contient les propriétés suivantes.

TABLEAU 5-3 Propriétés du groupe de propriétés `install_ipv4_interface`

Propriétés	Type	Requis	Description
<code>name</code>	<code>string</code>	<code>yes</code>	Nom de l'interface réseau.
<code>address_type</code>	<code>string</code>	<code>yes</code>	Valeur utilisée pour définir l'option <code>-T</code> pour la sous-commande <code>ipadm(1M) create-addr</code> . Les valeurs valides sont <code>static</code> ou <code>dhcp</code> .
<code>static_address</code>	<code>net_address_v4</code>	<code>no</code>	Requis uniquement avec un <code>address_type</code> <code>static</code> . Utilisé pour définir l'adresse locale pour la sous-commande <code>ipadm(1M) create-addr</code> .
<code>dhcp_wait</code>	<code>string</code>	<code>no</code>	S'applique uniquement à un <code>address_type</code> <code>dhcp</code> . Si elle est définie, cette propriété est utilisée pour définir la partie <code>-w secondes</code> (ou <code>forever</code> ) de la sous-commande <code>ipadm(1M) create-addr</code> .

TABLEAU 5-3 Propriétés du groupe de propriétés `install_ipv4_interface` (Suite)

Propriétés	Type	Requis	Description
<code>default_route</code>	<code>net_address_v4</code>	no	Utilisé pour définir une route par défaut à l'aide de la commande <code>route(1M)</code> .  <pre># /usr/sbin/route \ -p add default default-route \ -ifp ifname</pre> <p>La valeur de <i>ifname</i> est la partie du nom de l'interface de la propriété <i>name</i>.</p>

Le groupe de propriétés `install_ipv6_interface` contient les propriétés suivantes.

TABLEAU 5-4 Propriétés du groupe de propriétés `install_ipv6_interface`

Propriétés	Type	Requis	Description
<code>name</code>	<code>astring</code>	yes	Nom de l'interface réseau.
<code>address_type</code>	<code>astring</code>	yes	Valeur utilisée pour définir l'option <code>-T</code> pour la sous-commande <code>ipadm(1M) create-addr</code> . Les valeurs valides sont <code>static</code> ou <code>addrconf</code> .
<code>static_address</code>	<code>net_address_v6</code>	no	Requis uniquement avec un <code>address_type</code> <code>static</code> . Utilisé pour définir l'adresse locale pour la sous-commande <code>ipadm(1M) create-addr</code> .
<code>interface_id</code>	<code>net_address_v6</code>	no	S'applique uniquement à un <code>address_type</code> <code>addrconf</code> . Utilisé pour définir la partie <code>-i interface_id</code> de la sous-commande <code>ipadm(1M) create-addr</code> .
<code>stateless</code>	<code>astring</code>	no	S'applique uniquement à un <code>address_type</code> <code>addrconf</code> . Utilisé pour définir la partie <code>-p stateless=yes no</code> de la sous-commande <code>ipadm(1M) create-addr</code> .
<code>stateful</code>	<code>astring</code>	no	S'applique uniquement à un <code>address_type</code> <code>addrconf</code> . Utilisé pour définir la partie <code>-p stateful=yes no</code> de la sous-commande <code>ipadm(1M) create-addr</code> .

TABLEAU 5-4 Propriétés du groupe de propriétés `install_ipv6_interface` (Suite)

Propriétés	Type	Requis	Description
<code>default_route</code>	<code>net_address_v6</code>	no	Utilisé pour définir une route par défaut à l'aide de <code>route(1M)</code> .  <pre># /usr/sbin/route \ -p add default default-route \ -ifp ifname</pre> <p>La valeur de <code>ifname</code> est la partie du nom de l'interface de la propriété <code>name</code>.</p>

Le service `svc:/network/dns/install` prend en charge la configuration d'un client DNS. Le service définit un groupe de propriétés : `install_props`. Le service utilise ses propriétés pour créer un fichier `resolv.conf` (4) DNS.

Le groupe de propriétés `install_props` contient les propriétés suivantes.

TABLEAU 5-5 Propriétés du groupe de propriétés `install_props`

Propriétés	Type	Requis	Description
<code>domaine</code>	<code>astring</code>	no	Nom de domaine local. Utilisé pour définir la directive de domaine dans <code>resolv.conf</code> (4).
<code>nameserver</code>	<code>net_address</code>	yes	Utilisé pour définir les directives <code>nameserver</code> dans <code>resolv.conf</code> (4). La <code>net_address_list</code> doit contenir les adresses IPv4 et IPv6.
<code>search</code>	<code>astring</code>	no	Valeur pour l'ordre de recherche des noms d'hôte. Cette valeur est utilisée pour construire la directive de recherche dans <code>resolv.conf</code> (4). La <code>astring_list</code> doit contenir les valeurs de domaine.

## Exemples de fichiers manifest SC

Les exemples de cette section représentent des fichiers manifest SC terminés qui peuvent être intégrés dans un fichier manifest IA ou inclus par référence.

### Spécification du type de terminal et de la disposition du clavier

L'exemple de fichier manifest SC ci-dessous permet de configurer les paramètres suivants :

- Comptes utilisateur et root
- Fuseau horaire

- Nom d'hôte
- Type de terminal
- Configuration du clavier
- NWAM de configuration réseau

```

<?xml version='1.0'?>
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="system configuration">
  <service name="system/install/config" version="1" type="service">
    <instance name="default" enabled="true">
      <property_group name="user_account" type="application">
        <propval name="login" type="astring" value="jack"/>
        <propval name="password" type="astring" value="encrypted_password"/>
        <propval name="description" type="astring" value="default user"/>
        <propval name="shell" type="astring" value="/usr/bin/bash"/>
        <propval name="uid" type='count' value='101'/>
        <propval name="gid" type='count' value='10'/>
        <propval name="type" type="astring" value="normal"/>
        <propval name="roles" type="astring" value="root"/>
      </property_group>

      <property_group name="root_account" type="application">
        <propval name="password" type="astring" value="encrypted_password"/>
        <propval name="type" type="astring" value="role"/>
      </property_group>

      <property_group name="other_sc_params" type="application">
        <propval name="timezone" type="astring" value="GMT"/>
        <propval name="hostname" type="astring" value="solaris"/>
      </property_group>
    </instance>
  </service>

  <service name="system/console-login" version="1" type="service">
    <property_group name="ttymon" type="application">
      <propval name="terminal_type" type="astring" value="sun"/>
    </property_group>
  </service>

  <service name='system/keymap' version='1' type='service'>
    <instance name='default' enabled='true'>
      <property_group name='keymap' type='system'>
        <propval name='layout' type='astring' value='US-English'/>
      </property_group>
    </instance>
  </service>

  <service name="network/physical" version="1" type="service">
    <instance name="nwam" enabled="true"/>
    <instance name="default" enabled="false"/>
  </service>
</service_bundle>

```



## Spécification de la configuration réseau statique

L'exemple de fichier manifest SC ci-dessous permet de configurer les paramètres suivants :

- bge0 avec l'adresse IPv4 statique 10.0.0.10, le masque de réseau 255.0.0.0
- Route par défaut IPv4 10.0.0.1
- bge1 avec type d'adresse addrconf IPv6
- Serveur de noms DNS 8.8.8.8
- example1.com en tant que nom de domaine DNS local
- example2.com, example3.com en tant que liste de recherche DNS pour la recherche de nom d'hôte

```
<?xml version='1.0'?>
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="system configuration">
  <service name="system/install/config" version="1" type="service">
    <instance name="default" enabled="true">
      <property_group name="user_account" type="application">
        <propval name="login" type="astring" value="jack"/>
        <propval name="password" type="astring" value="encrypted_password"/>
        <propval name="description" type="astring" value="default user"/>
        <propval name="shell" type="astring" value="/usr/bin/bash"/>
        <propval name="uid" type='count' value='101'/>
        <propval name="gid" type='count' value='10'/>
        <propval name="type" type="astring" value="normal"/>
        <propval name="roles" type="astring" value="root"/>
      </property_group>

      <property_group name="root_account" type="application">
        <propval name="password" type="astring" value="encrypted_password"/>
        <propval name="type" type="astring" value="role"/>
      </property_group>

      <property_group name="other_sc_params" type="application">
        <propval name="timezone" type="astring" value="GMT"/>
        <propval name="hostname" type="astring" value="solaris"/>
      </property_group>
    </instance>
  </service>

  <service name="system/console-login" version="1" type="service">
    <property_group name="ttymon" type="application">
      <propval name="terminal_type" type="astring" value="sun"/>
    </property_group>
  </service>

  <service name='system/keymap' version='1' type='service'>
    <instance name='default' enabled='true'>
      <property_group name='keymap' type='system'>
        <propval name='layout' type='astring' value='US-English'/>
      </property_group>
    </instance>
  </service>
```

```
<service name="network/physical" version="1" type="service">
  <instance name="nwam" enabled="false"/>
  <instance name="default" enabled="true"/>
</service>

<service name='network/install' version='1' type='service'>
  <instance name='default' enabled='true'>
    <property_group name='install_ipv4_interface' type='application'>
      <propval name='name' type='astring' value='bge0/v4' />
      <propval name='address_type' type='astring' value='static' />
      <propval name='static_address' type='net_address_v4' value='10.0.0.10/8' />
      <propval name='default_route' type='net_address_v4' value='10.0.0.1' />
    </property_group>

    <property_group name='install_ipv6_interface' type='application'>
      <propval name='name' type='astring' value='bge1/v6' />
      <propval name='address_type' type='astring' value='addrconf' />
      <propval name='stateless' type='astring' value='yes' />
      <propval name='stateful' type='astring' value='yes' />
    </property_group>
  </instance>
</service>

<service name='network/dns/install' version='1' type='service'>
  <instance name='default' enabled='true'>
    <property_group name='install_props' type='application'>
      <property name='nameserver' type='net_address'>
        <net_address_list>
          <value_node value='8.8.8.8' />
        </net_address_list>
      </property>
      <property name='domain' type='astring' value='example1.com' />
      <property name='search' type='astring'>
        <astring_list>
          <value_node value='example2.com' />
          <value_node value='example3.com' />
        </astring_list>
      </property>
    </property_group>
  </instance>
</service>
</service_bundle>
```

# Configuration de DHCP pour le programme d'installation automatisée

---

Le programme d'installation automatisée utilise DHCP pour indiquer l'adresse IP, le masque de sous-réseau, le routeur, le serveur DNS et l'emplacement d'une image d'initialisation à l'ordinateur client à installer.

- Si vous souhaitez configurer un serveur DHCP Oracle Solaris ou le configurer pour une utilisation avec le programme d'installation automatisée, reportez-vous à la section [“DHCP Oracle Solaris pour le programme d'installation automatisée”](#) à la page 83.
- Si vous souhaitez configurer un serveur DHCP ISC ou le configurer pour une utilisation avec le programme d'installation automatisée, reportez-vous à la section [“ISC DHCP pour le programme d'installation automatisée”](#) à la page 86.

## DHCP Oracle Solaris pour le programme d'installation automatisée

Vous pouvez configurer votre serveur d'installation IA afin qu'il soit également un serveur DHCP Oracle Solaris, ou vous pouvez configurer un serveur DHCP distinct.

Si vous avez déjà configuré un serveur DHCP Oracle Solaris, vous devrez peut-être le configurer pour qu'il fonctionne avec le programme d'installation automatisée.

## Configuration d'un serveur DHCP sur le serveur d'installation IA

Si vous n'avez pas configuré un serveur DHCP, vous pouvez utiliser la commande `installadm create-service` avec les options `-i` et `-c` pour configurer un pool d'adresses IP à utiliser avec ce service d'installation. Reportez-vous à la section [“Création d'un service d'installation comprenant la configuration du serveur DHCP Oracle Solaris”](#) à la page 22 pour obtenir un

exemple. Consultez également la page de manuel `installadm(1M)`. L'option `-i` définit un nouveau serveur DHCP à partir de l'adresse IP spécifiée. L'option `-c` définit le nombre spécifié d'adresses IP dans la table DHCP.

## Configuration d'un serveur DHCP distinct

Si vous souhaitez configurer un serveur DHCP Oracle Solaris distinct, reportez-vous à la section [Partie II, "DHCP" du \*System Administration Guide: IP Services\*](#). Passez ensuite à la configuration pour le programme d'installation automatisée décrite dans ["Configuration d'un serveur DHCP existant pour le programme d'installation automatisée"](#) à la page 84.

## Configuration d'un serveur DHCP existant pour le programme d'installation automatisée

Si vous n'utilisez pas les options `-i` et `-c`, la commande `installadm create-service` fournit des instructions vous permettant de configurer DHCP. Cette section fournit des informations supplémentaires sur la configuration d'un serveur DHCP pour le programme d'installation automatisée.

### Configuration de la table Netmasks

Assurez-vous que les entrées de la table `netmasks` du serveur DHCP sont correctes, pour tous les réseaux pour lesquels il fournira le service DHCP.

```
# cat >>/etc/netmasks
192.168.0.0 255.255.255.0
# getent netmasks 192.168.0.0
192.168.0.0 255.255.255.0
```

### Initialisation du service de serveur DHCP

Exécutez `dhcpconfig(1M)` pour initialiser le service SMF DHCP. La commande suivante crée un référentiel basé sur des fichiers et une macro d'hôte avec un bail par défaut d'un jour. Cette commande active également le service SMF `svc:/network/dhcp-server`.

```
# /usr/sbin/dhcpconfig -D -r SUNWfiles -p /var/dhcp
# dhtadm -P
Name          Type      Value
=====
example-host  Macro    :Include=Locale:DNSdomain="example.com":DNSserv=192.168.0.1:
Locale        Macro    :UTCoffst=-25200:
```

## Ajout de macros et de tables d'information réseau

Pour ajouter des macros pour chaque réseau de site, utilisez `dhcpconfig` pour ajouter la macro de site et renseigner les tables de réseau. L'option `-t` indique le routeur pour ce réseau. Vous pouvez également utiliser cette fonction pour ajouter de nouvelles informations réseau aux serveurs DHCP hébergeant les réseaux existants.

```
# dhcpconfig -N 192.168.0.1 -m 255.255.255.0 -t 192.168.0.1
```

## Création et ajout d'une macro DHCP pour le programme d'installation automatisée

Pour configurer votre serveur DHCP Oracle Solaris en vue d'une utilisation avec le programme d'installation automatisée, créez une macro DHCP et utilisez `dhtadm` pour ajouter la macro à la table de configuration DHCP, `dhcptab(4)`. Voir l'exemple de sortie `installadm create-service` dans ["Création d'un service d'installation sans configuration DHCP"](#) à la page 21.

La première macro créée pour une architecture est la macro par défaut pour tous ses clients.

## Affectation des adresses IP de client

Affectez les adresses IP de client si nécessaire.

Utilisez la commande `pntadm(1M)` pour ajouter les réseaux à la table de réseau DHCP, `dhcp_network(4)`.

Vérifiez les tables de réseau :

```
# pntadm -L
```

## Attribution d'un bail DHCP à un client

Attribuez au client un bail DHCP et définissez sa macro DHCP pour utiliser l'image d'installation IA correcte. Exécutez les commandes de cette section sur le serveur DHCP.

### Attribution du bail DHCP

L'option `-i` de la commande `pntadm` est l'ID du client à installer. L'option `-m` spécifie le nom de cette macro. Dans cet exemple, le nom de la macro est le même que l'adresse MAC du client. Ceci permet d'assurer que vous n'avez qu'un bail par hôte.

```
# pntadm -A 192.168.0.2 -i 01C0FFEEC0FFEE -m 01C0FFEEC0FFEE \  
-f "PERMANENT+MANUAL" 192.168.0.0
```

### Vérification de la présence du bail

La valeur d'indicateurs `03` représente `PERMANENT+MANUAL` : le bail est permanent et le serveur DHCP ne sera pas en mesure d'attribuer une adresse différente pour le client.

```
# pntadm -P 192.168.0.0 | grep 01C0FFEEC0FFEE
Client ID      Flags    Client IP    Server IP    Lease Expiration  Macro          Comment
01C0FFEEC0FFEE 03      10.41.30.42  172.30.95.10 Forever      01C0FFEEC0FFEE
```

## Allocation de la macro

```
# dhtadm -g -A -m 01C0FFEEC0FFEE \  
-d "":Include='uname -n':BootSrvA=192.168.0.1:BootFile=install_test_ai_x86:"
```

## Vérification de la présence de la nouvelle macro dans la table

```
# dhtadm -P |grep 01C0FFEEC0FFEE
01C0FFEEC0FFEE Macro :Include=example-host:BootSrvA=192.168.0.1:BootFile="install_test_ai_x86":
```

# ISC DHCP pour le programme d'installation automatisée

Cette section décrit la configuration DHCP ISC pour servir les informations dont vous avez besoin pour les installations IA.

## Configuration réseau de base

Définissez un réseau de base dans votre fichier `dhcpd.conf`. Dans l'exemple suivant, le réseau 192.168.0.0/24 est défini afin de servir les adresses IP comprises entre 2 et 100 avec un routeur de 192.168.0.1 :

```
subnet 192.168.0.0 netmask 255.255.255.0 {  
    range 192.168.0.2 192.168.0.100;  
    option routers 192.168.0.1;  
}
```

## Configuration de DNS

De même, ajoutez les informations DNS à votre fichier `dhcpd.conf`. Si vous souhaitez que chaque sous-réseau servi obtienne des informations DNS différentes, placez les directives dans le bloc de sous-réseau ou, pour que tous les sous-réseaux servis soient affectés, au début du fichier.

```
option domain-name "example.com";  
option domain-name-servers 192.168.0.1;
```

## Configuration du serveur d'initialisation et du fichier d'initialisation

Le programme d'installation automatisée nécessite une configuration du serveur d'initialisation et du fichier d'initialisation dans DHCP. Lorsque vous démarrez un ordinateur client pour l'installation automatisée, ce client a besoin de connaître l'emplacement à partir duquel il peut obtenir un fichier d'initialisation (à partir du serveur IA) et le nom du fichier d'initialisation. Ces informations sont fournies par la commande `installadm create-service` lorsque vous configurez un service IA. L'exemple suivant illustre la sortie partielle d'une commande `installadm create-service` :

```
Boot server IP (BootSrvA) : 192.168.0.1
Boot file      (BootFile)  : install_test_ai_x86
GRUB Menu     (GrubMenu)  : menu.lst.install_test_ai_x86
```

En termes de DHCP ISC, le serveur d'initialisation est la directive `next-server`, et le fichier d'initialisation est la directive `filename`. Ajoutez ces directives à votre groupe de sous-réseau, comme indiqué dans l'exemple :

```
subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.2 192.168.0.100;
    option routers 192.168.0.1;
    filename "install_test_ai_x86";
    next-server 192.168.0.1;
}
```

Pour un service IA SPARC, utilisez la même directive `filename` pour l'objet `BootFile`. Les installations SPARC ne nécessitent pas de directive `next-server` (`BootSrvA`).

Si vous utilisez le programme d'installation automatisée pour installer les machines x86 et SPARC, utilisez la directive DHCP ISC `class` pour fournir les informations de fichier d'initialisation aux clients SPARC et fournir les informations de fichier d'initialisation et de serveur d'initialisation aux clients x86. Le fait de fournir des informations spécifiques à l'initialisation dans des directives `class` distinctes vous permet de disposer d'un service par défaut pour chaque architecture sur le réseau.

### Classe d'initialisation PXE

L'exemple suivant est une définition `class` pour une initialisation matérielle x86 :

```
class "PXEBoot" {
    option dhcp-class-identifiant "PXEClient";
    filename "install_test_ai_x86";
    next-server 192.168.0.1;
}
```

## Classe d'initialisation SPARC

L'exemple suivant est une définition `class` pour l'initialisation matérielle SPARC. Notez qu'une directive `next-server` n'est pas nécessaire pour SPARC :

```
class "SPARC" {
    match if ( substring (option vendor-class-identifiant, 0, 5) = "SUNW." ) and not
              ( option vendor-class-identifiant = "SUNW.i86pc" );
    filename "http://192.168.0.1:5555/cgi-bin/wanboot-cgi";
}
```

## Exemple de fichier de configuration

Avec ces définitions de classe, les clients SPARC demandent un bail et obtiennent des informations spécifiques à SPARC, tandis que les clients x86 demandent et obtiennent des informations spécifiques à x86. Les exemples ci-après présentent l'intégralité du fichier `dhcpd.conf` :

```
# option definitions common to all supported networks...
option domain-name "example.com";
option domain-name-servers 192.168.0.1;

default-lease-time 600;
max-lease-time 86400;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

# This is an easy way to discriminate on SPARC clients
class "SPARC" {
    match if ( substring (option vendor-class-identifiant, 0, 5) = "SUNW." ) and not
              ( option vendor-class-identifiant = "SUNW.i86pc" );
    filename "http://192.168.0.1:5555/cgi-bin/wanboot-cgi";
}

# This is a class to discriminate on PXE booting x86 clients
class "PXEBoot" {
    option dhcp-class-identifiant "PXEClient";
    filename "install_test_ai_x86";
    next-server 192.168.0.1;
}

# This is a very basic subnet declaration
subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.2 192.168.0.100;
    option routers 192.168.0.1;
}
```



# Installation de systèmes clients

---

L'installation démarre lorsque vous démarrez le client. Lorsque le client démarre, DHCP dirige le client vers le serveur d'installation IA.

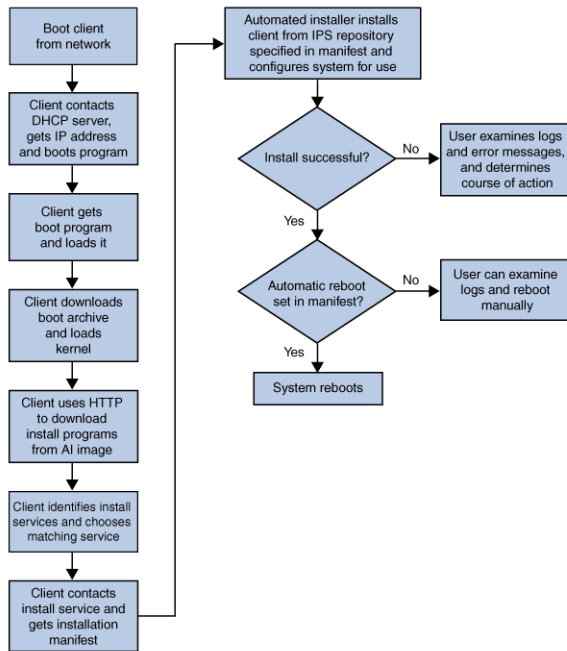
Ce chapitre indique la configuration système requise pour les clients d'installation automatisée et décrit l'association de chaque client avec le service d'installation approprié.

## Installation d'un client

Lorsque de la configuration de votre serveur d'installation, vous avez créé au moins un service d'installation pour chaque architecture de client et chaque version du SE Oracle Solaris que vous envisagez d'installer. Lorsque vous avez créé chaque service d'installation, vous avez créé des instructions d'installation personnalisée et des instructions de configuration post-installation pour différents clients, selon les besoins. Pour démarrer l'installation automatisée, il suffit d'initialiser le client.

L'organigramme suivant illustre la manière dont un système client est installé. Lors de l'initialisation, le client utilise les spécifications dans les arguments d'initialisation pour identifier le service d'installation correct à utiliser. L'installation est effectuée sur le système client à l'aide d'une image d'initialisation, des spécifications d'installation et des spécifications de configuration post-installation fournies par le service d'installation.

FIGURE 7-1 Étapes d'installation du client



## Configuration système requise pour les clients

Les systèmes client doivent correspondre à la configuration requise suivante pour l'installation automatisée. Tous les systèmes qui correspondent à cette configuration requise peuvent être utilisés en tant que clients d'installation automatisée, y compris les ordinateurs portables, les ordinateurs de bureau, les machines virtuelles et les serveurs d'entreprise.

## Configuration système requise pour les clients SPARC et x86

Les clients SPARC et x86 de l'installation IA sur le réseau doivent répondre aux exigences suivantes :

**Mémoire**                      **x86** : 512 Mo minimum

**SPARC** : 1 Go minimum

La mémoire minimale requise est de 1,5 Go si le fichier manifest IA spécifie l'installation du pilote périphérique. Reportez-vous à la section [“Identification et installation de pilotes manquants sur une cible](#)

d'installation” à la page 63 pour obtenir des informations sur l'installation automatisée du pilote de périphérique.

**Espace disque**    **SPARC et x86** : 13 Go minimum

**Accès réseau** : les systèmes client doivent être en mesure d'accéder aux ressources suivantes au cours de l'installation :

- un serveur DHCP fournissant des informations de configuration réseau ;
- le serveur d'installation IA ;
- un référentiel IPS contenant les packages à installer sur le système client.

## Configuration système requise supplémentaire pour les clients SPARC

### Initialisation via connexion WAN

Les clients SPARC de l'installation IA sur le réseau doivent prendre en charge l'initialisation via une connexion WAN. Reportez-vous à [“Mon client SPARC prend-il en charge l'initialisation via une connexion WAN ?”](#) à la page 91

### Microprogramme

Le microprogramme sur les clients SPARC doit être mis à jour pour inclure la version actuelle d'Open Boot PROM (OBP) qui contient la dernière prise en charge de l'initialisation via une connexion WAN.

## Mon client SPARC prend-il en charge l'initialisation via une connexion WAN ?

Afin d'initialiser sur le réseau, le programme d'installation automatisée requiert la prise en charge de l'initialisation via une connexion WAN pour les clients SPARC. Pour vous assurer que l'OBP du client prend en charge l'initialisation via une connexion WAN, vous pouvez vérifier si `network-boot-arguments` est une variable valide pouvant être définie dans `eeprom`.

Si la variable `network-boot-arguments` s'affiche, ou si la commande renvoie la sortie `network-boot-arguments: data not available`, l'OBP prend en charge l'initialisation via une connexion WAN, et le client peut être installé sur le réseau.

```
# eeprom | grep network-boot-arguments
network-boot-arguments: data not available
```

Si la commande n'affiche aucune sortie, l'initialisation via une connexion WAN n'est pas prise en charge, et le client ne peut pas être installé sur le réseau. Reportez-vous au [Chapitre 8](#), [“Installations automatisées initialisées à partir d'un support”](#).

```
# eeprom | grep network-boot-arguments
```

## Configuration d'un client d'installation

Sur le serveur d'installation, utilisez la commande `installadm create-client` pour associer un client spécifique à un service d'installation spécifique.

La commande `installadm create-client` requiert les informations suivantes :

- adresse MAC du client ;
- nom du service d'installation que le client doit utiliser pour l'installation.

(facultatif) propriétés d'initialisation pour les clients x86.

## Configuration d'un client x86

L'exemple suivant associe le client x86 avec l'adresse MAC `0:e0:81:5d:bf:e0` au service d'installation `s11-x86`. L'option `-b` définit la valeur de la console, `console=ttya`, dans le fichier `menu.lst` spécifique au client dans `/tftpboot`.

```
# installadm create-client -b "console=ttya" \
-e 0:e0:81:5d:bf:e0 -n s11-x86
Setting up X86 client...
Service discovery fallback mechanism set up

Detected that DHCP is not set up on this server.
If not already configured, please create a DHCP macro
named 0100E0815DBFE0 with:
  Boot server IP (BootSrvA) : 10.6.68.29
  Boot file      (BootFile)  : 0100E0815DBFE0
If you are running the Solaris DHCP server, use the following
command to add the DHCP macro, 0100E0815DBFE0:
  /usr/sbin/dhtadm -g -A -m 0100E0815DBFE0 -d \
:BootSrvA=10.6.68.29:BootFile=0100E0815DBFE0:GrubMenu=menu.lst.0100E0815DBFE0:
```

Note: Be sure to assign client IP address(es) if needed (e.g., if running the Solaris DHCP server, run `pntadm(1M)`).

Cette commande affiche le nom et les valeurs d'une macro, `0100E0815DBFE0`, qui doit être ajoutée au serveur DHCP.

Utilisez l'utilitaire `/usr/smap/admin/bin/dhcpmgr` pour ouvrir le gestionnaire DHCP afin d'afficher les résultats de la configuration DHCP dans la table DHCP.

Vous pouvez également afficher les résultats de la commande `installadm create-client` dans le répertoire `/tftpboot` :

```

# cd /tftpboot
# ls -l
lrwxrwxrwx 13:23 0100E0815DBFE0 -> pxegrub.I86PC.Solaris-1
drwxr-xr-x 13:26 I86PC.Solaris-1
-rw-r--r-- 13:23 menu.lst.0100E0815DBFE0
-rwxr-xr-x 13:23 pxegrub.I86PC.Solaris-1
-rw-r--r-- 13:23 rm.0100E0815DBFE0

# cat menu.lst.0100E0815DBFE0
default=0
timeout=30
min_mem64=1000
title Oracle Solaris 11 Express boot image
    kernel$ /I86PC.Solaris-1/platform/i86pc/kernel/$ISADIR/unix
-B install_media=http://135.134.0.10:5555/export/aiserver/s11-ai-x86,
install_service=s11-ai-x86,install_svc_address=135.134.0.10:46501
    module$ /I86PC.Solaris-1/platform/i86pc/$ISADIR/boot_archive
title Oracle Solaris 11 Express Automated Install
    kernel$ /I86PC.Solaris-1/platform/i86pc/kernel/$ISADIR/unix
-B install=true,install_media=http://135.134.0.10:5555/export/aiserver/s11-ai-x86,
install_service=osol-1003-ai-x86,install_svc_address=135.134.0.10:46501
    module$ /I86PC.Solaris-1/platform/i86pc/$ISADIR/boot_archive

```

## Configuration d'un client SPARC

L'exemple suivant associe le client SPARC avec l'adresse MAC `00:14:4f:A7:65:70` au service d'installation `s11-sparc`.

```

# installadm create-client -e 00:14:4f:a7:65:70 -n s11-sparc
Setting up SPARC client...
Creating SPARC configuration file

Detected that DHCP is not set up on this server.
If not already configured, please create a DHCP macro
named 0100E0815DBFE0 with:
    Boot server IP (BootSrvA) : 10.6.68.29
    Boot file      (BootFile) : http://10.6.68.29:5555/cgi-bin/wanboot-cgi
If you are running the Oracle Solaris DHCP server, use the following
command to add the DHCP macro, 0100E0815DBFE0:
    /usr/sbin/dhtadm -g -A -m 0100E0815DBFE0 \
-d :BootSrvA=10.6.68.29:BootFile="http://10.6.68.29:5555/cgi-bin/wanboot-cgi":
Note: Be sure to assign client IP address(es) if needed
(e.g., if running the Oracle Solaris DHCP server, run pntadm(1M)).

```

Cette commande affiche le nom et les valeurs d'une macro, `0100E0815DBFE0`, qui doit être ajoutée au serveur DHCP. Si vous utilisez le serveur DHCP Oracle Solaris, vous pouvez créer la macro sur votre serveur DHCP à l'aide du gestionnaire DHCP ou en exécutant les commandes `dhtadm`, fournies dans les instructions, sur le serveur DHCP.

## Suppression d'un client à partir d'un service

Utilisez la commande `installadm delete-client` pour supprimer un client à partir d'un service d'installation.

```
# installadm delete-client -e macaddr
```

Vous n'avez pas besoin de spécifier le nom de service car un client ne peut être associé qu'à un seul service d'installation.

## Installation de clients

Initialisez le client pour démarrer l'installation. Cette section vous montre exactement comment initialiser un client SPARC ou x86. Cette section décrit également comment surveiller à distance la progression de l'installation.

## Utilisation de Secure Shell pour contrôler à distance les installations

Vous pouvez activer l'accès réseau à un client d'installation automatisée en utilisant `ssh`. Cet accès permet d'observer à distance un processus d'installation.

Activez-le en définissant l'option `livessh` sur `enable` dans le fichier de configuration d'installation. Lorsque cet accès est activé, vous pouvez vous connecter au client IA à l'aide du nom d'utilisateur et du mot de passe, `jack`.

### Contrôle à distance des installations de client x86

Pour les systèmes x86, le fichier de configuration `menu.lst` est créé dans le répertoire `/tftpboot/` avec l'un des formats de nom de fichier suivants.

- Si vous avez utilisé la commande `installadm create-client`, le nom de fichier est `menu.lst.01adresse_MAC`, où `adresse_MAC` correspond à l'adresse MAC qui a été spécifiée dans la commande `installadm create-client`.
- Si vous n'avez pas utilisé la commande `installadm create-client`, le nom de fichier est `menu.lst.nom_service`, où `nom_service` est le nom du service d'installation qui a été spécifié dans la commande `installadm create-service`.

Dans ce fichier, les options sont fournies en tant que paramètres du noyau. Dans l'exemple suivant, les options `livessh` et `install_debug` sont définies sur `enable`.

```
kernel$ ... -B install_media=...,livessh=enable,install_debug=enable
```

## Contrôle des installations de clients SPARC

Pour les systèmes SPARC, le fichier `install.conf` est créé dans le répertoire `ai_image_dest` lors de la création d'un service d'installation à l'aide d'une commande `installadm create-service`.

Dans le fichier `install.conf`, les options sont définies en tant que paires nom-valeur. Dans l'exemple suivant, l'option `livessh` est définie sur `enable`.

```
$ cat ai_image_dest/install.conf
...
livessh=enable
...
```

## Installation d'un client SPARC

Utilisez la commande suivante pour démarrer les clients SPARC à partir de l'invite OBP :

```
OK boot net:dhcp - install
```

## Séquence d'initialisation réseau des clients SPARC

Les événements suivants se produisent pendant l'initialisation IA d'un client SPARC :

1. Le client est initialisé et récupère l'adresse IP et le fichier d'initialisation `wanboot-cgi` à partir du serveur DHCP.
2. Le fichier d'initialisation `wanboot-cgi` lit `wanboot.conf` et envoie l'emplacement du binaire d'initialisation via une connexion WAN au client.
3. Le binaire d'initialisation via la connexion WAN est téléchargé à l'aide du protocole HTTP, et le client initialise le programme d'initialisation via la connexion WAN.
4. Le programme d'initialisation via une connexion WAN obtient le fichier `boot_archive` et le SE Oracle Solaris est initialisé.
5. Les archives d'image `solaris.zlib` et `solarismisc.zlib` sont téléchargées via le protocole HTTP.
6. Le service SMF IA utilise le protocole Multicast DNS (mDNS) afin de contacter le serveur HTTP et d'obtenir le fichier `manifest IA`.
7. Le programme d'installation IA est appelé à l'aide du fichier `manifest IA` pour effectuer l'installation du SE Oracle Solaris sur le client.

## Exemple de sortie d'initialisation réseau SPARC

La séquence d'initialisation affiche la sortie suivante :

```
{3} ok boot net:dhcp - install

...
OpenBoot 4.23.4, 8184 MB memory available, Serial #69329298.
Ethernet address 0:14:4f:21:e1:92, Host ID: 8421e192.

Rebooting with command: boot net:dhcp - install
Boot device: /pci@7c0/pci@0/network@4:dhcp File and args:
1000 Mbps FDX Link up
1000 Mbps FDX Link up
<time unavailable> wanboot info: WAN boot messages->console
<time unavailable> wanboot info: Starting DHCP configuration
<time unavailable> wanboot info: DHCP configuration succeeded
<time unavailable> wanboot progress: wanbootfs: Read 366 of 366 kB (100%)
<time unavailable> wanboot info: wanbootfs: Download complete
Tue Aug 5 17:12:09 wanboot progress: miniroot: Read 165251 of 165251 kB (100%)
Tue Aug 5 17:12:09 wanboot info: miniroot: Download complete
SunOS Release 5.11 Version snv_151 64-bit
...
Hostname: solaris
Remounting root read/write
Probing for device nodes ...
Preparing automated install image for use
Downloading solaris.zlib archive
--11:09:11-- http://10.6.35.226:5555//export/home \
/images/s11-ai-sparc//solaris.zlib
=> '/tmp/solaris.zlib'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 200 OK
Length: 82,679,296 (79M) [text/plain]

100%[=====] 82,679,296 33.39M/s

11:09:13 (33.36 MB/s) - '/tmp/solaris.zlib' saved [82679296/82679296]

Downloading solarismisc.zlib archive
--11:09:13-- http://10.6.35.226:5555//export/home/images \
/s11-ai-sparc//solarismisc.zlib
=> '/tmp/solarismisc.zlib'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 200 OK
Length: 620,032 (606K) [text/plain]

100%[=====] 620,032 ---K/s

11:09:13 (36.48 MB/s) - '/tmp/solarismisc.zlib' saved [620032/620032]

--11:09:13-- http://10.6.35.226:5555//export/home/images/ \
s11-ai-sparc//install.conf
=> '/tmp/install.conf'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 200 OK
Length: 39 [text/plain]

100%[=====] 39 ---K/s

11:09:13 (953.58 KB/s) - '/tmp/install.conf' saved [39/39]
```



```
Done mounting automated install image
Configuring devices.
Reading ZFS config: done.
```

```
Automated Installation started
The progress of the Automated Installation can be followed by
viewing the logfile at /tmp/install_log
```

## Installation d'un client x86

Lancez l'installation du client x86 à l'aide de l'une des méthodes suivantes pour démarrer à partir du réseau :

- Appuyez sur la touche de fonction appropriée. Par exemple, certains systèmes utilisent F12 pour démarrer à partir du réseau.
- Changez l'ordre d'initialisation dans le BIOS.

Lors de l'initialisation du client, sélectionnez le périphérique réseau d'initialisation.

## Séquence d'initialisation réseau des clients x86

Les événements suivants se produisent pendant l'initialisation IA d'un client x86 :

1. Le client est initialisé et obtient une adresse IP, et le fichier d'initialisation, `pxegrub`, est téléchargé depuis l'emplacement fourni par le serveur DHCP.
2. Le fichier d'initialisation `pxegrub` est chargé et lit un fichier `menu.lst`.
3. Le fichier d'initialisation `pxegrub` obtient le fichier `boot_archive` et le SE Oracle Solaris est démarré en utilisant TFTP.
4. Les archives d'image réseau, `solaris.zlib` et `solarismisc.zlib`, sont téléchargées par le biais du protocole HTTP tel que proposé par le menu GRUB.
5. Le fichier manifest IA est téléchargé depuis un serveur HTTP indiqué dans le menu GRUB ou à partir de la recherche mDNS effectuée par le service SMF IA.
6. Le programme d'installation IA est appelé à l'aide du fichier manifest IA pour effectuer l'installation.

## Exemple de sortie d'initialisation réseau x86

Le premier écran affiche le message suivant lorsque le client reçoit la réponse DHCP correcte :

```
Intel(R) Boot Agent PXE Base Code (PXE-2.1 build 0.86)
Copyright(C) 1997-2007, Intel Corporation
```

```
CLIENT MAC ADDR 00 14 4F 29 04 12 GUID FF2000008 FFFF FFFF FFFF 7BDA264F1400
CLIENT IP: 10.6.68.29 MASK: 255.255.255.0 DHCP IP: 10.6.68.49
GATEWAY: 10.6.68.1
```

Le menu GRUB s'affiche avec deux entrées de menu. Sélectionnez la seconde entrée pour lancer une installation automatisée :

```
Oracle Solaris 11 Express boot image
Oracle Solaris 11 Express Automated Install
```

L'entrée de menu GRUB par défaut, "Oracle Solaris 11 Express boot image" (image d'initialisation d'Oracle Solaris 11 Express), initialise l'image sans démarrer une installation automatisée "mains libres". Sélectionnez la deuxième entrée dans le menu GRUB, "Oracle Solaris 11 Express Automated Install" pour lancer une installation automatisée.

Une fois l'image sélectionnée, les messages suivants s'affichent :

```
SunOS Release 5.11 Version snv_151 64-bit
...
Hostname: solaris
Remounting root read/write
Probing for device nodes ...
Preparing automated install image for use
Downloading solaris.zlib archive
--11:09:11-- http://10.6.35.226:5555//export/home \
/images/s11-ai-x86//solaris.zlib
=> '/tmp/solaris.zlib'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 200 OK
Length: 82,679,296 (79M) [text/plain]

100%[=====>] 82,679,296    33.39M/s

11:09:13 (33.36 MB/s) - '/tmp/solaris.zlib' saved [82679296/82679296]

Downloading solarismisc.zlib archive
--11:09:13-- http://10.6.35.226:5555//export/home/images \
/s11-ai-x86//solarismisc.zlib
=> '/tmp/solarismisc.zlib'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 200 OK
Length: 620,032 (606K) [text/plain]

100%[=====>] 620,032    ---K/s

11:09:13 (36.48 MB/s) - '/tmp/solarismisc.zlib' saved [620032/620032]

--11:09:13-- http://10.6.35.226:5555//export/home/images/ \
s11-ai-x86//install.conf
=> '/tmp/install.conf'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 200 OK
Length: 39 [text/plain]

100%[=====>] 39    ---K/s

11:09:13 (953.58 KB/s) - '/tmp/install.conf' saved [39/39]

Done mounting automated install image
```

```
Configuring devices.  
Reading ZFS config: done.
```

```
Automated Installation started  
The progress of the Automated Installation can be followed by  
viewing the logfile at /tmp/install_log
```

## Messages d'installation du client

Les messages suivants sont communs aux deux installations SPARC et x86.

### Message d'installation automatisée démarrée

Si le client parvient à s'initialiser et à télécharger les fichiers d'installation, le message suivant s'affiche :

```
Automated Installation started  
  The progress of the Automated Installation can be followed by viewing the  
  logfile at /tmp/install_log
```

Vous pouvez vous connecter en tant que `root` avec le mot de passe `solaris` pour surveiller les messages d'installation dans `/tmp/install_log`. Une fois que l'installation des packages à partir d'IPS a démarré, ce fichier journal ne peut être mis à jour pendant une période prolongée.

### Message d'installation automatisée réussie

Si le message suivant s'affiche, l'installation est terminée :

```
Auto install succeeded. You may wish to reboot the system at this time.
```

Si vous avez configuré une réinitialisation automatisée dans le fichier manifest IA, le système est réinitialisé à ce stade. Pour spécifier le redémarrage automatique après une installation réussie, définissez l'attribut `auto_reboot` de la balise `<ai_instance>` sur `true`. La valeur par défaut est `false`, ce qui signifie que le client n'est pas réinitialisé automatiquement après une installation réussie.



# Installations automatisées initialisées à partir d'un support

---

Vous pouvez lancer une installation automatisée du système d'exploitation Oracle Solaris 11 Express sur un système SPARC ou x86 en initialisant une image IA sur un support plutôt qu'en initialisant sur le réseau. Ce chapitre traite des motifs d'initialisation d'un client IA à partir d'un support et de la procédure d'installation dans ce mode.

## Présentation de l'installation effectuée à partir d'un support IA

L'installation à partir d'un support IA vous permet d'accomplir les tâches suivantes :

- Installation du système qui sera votre serveur d'installation IA.
- Installation d'un système SPARC qui ne permet pas l'initialisation via connexion WAN.
- Dépannage d'un système en difficulté. Initialisation du système à partir du support amovible, puis vérification du système installé et exécution des tests de diagnostic.

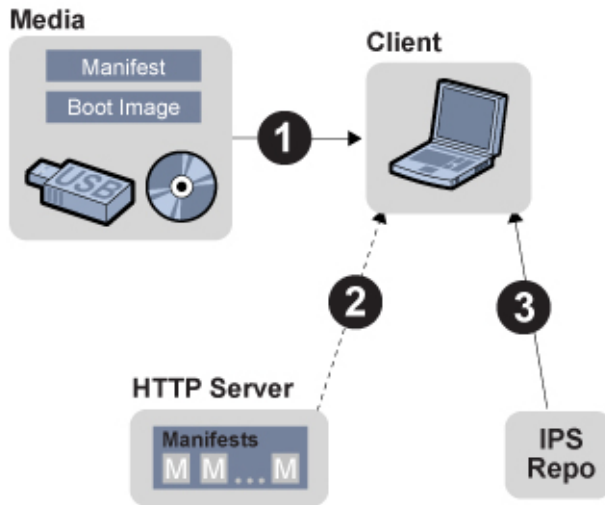
L'installation à partir d'un support IA possède les caractéristiques suivantes :

- Vous n'avez pas besoin de configurer un serveur d'installation ou un service d'installation.
- Le système n'a pas besoin d'être en mesure de démarrer sur le réseau.

## Installation à partir d'un support IA

Vous pouvez initialiser une image IA à partir d'un CD, d'un DVD ou d'un périphérique USB pour lancer une installation mains-libres de ce système uniquement. Un fichier manifest IA fournit des instructions d'installation et de configuration. Le système à installer doit disposer d'un accès réseau. Pour terminer l'installation, les packages logiciels sont récupérés à partir d'un référentiel IPS sur Internet ou sur le réseau local. Vérifiez le fichier manifest IA par défaut comme décrit dans la section [“Création d'un fichier manifest IA personnalisé”](#) à la page 103.

FIGURE 8-1 Installation IA à partir d'un support



## Configuration système requise pour l'installation à partir d'un support IA

Les systèmes SPARC et x86 doivent satisfaire aux conditions suivantes.

TABLEAU 8-1 Configuration système requise pour l'installation à partir d'un support IA

Configuration requise	Spécifications
Mémoire	Mémoire minimum : 512 Mo
Espace disque	Espace disque minimum : 13 Go
Accès réseau	<p>Le système à installer doit être en mesure d'accéder aux ressources suivantes au cours de l'installation :</p> <ul style="list-style-type: none"> <li>▪ un serveur DHCP fournissant des informations de configuration réseau ;</li> <li>▪ un référentiel IPS contenant les packages à installer sur le système client.</li> </ul> <p>Si vous créez un fichier manifest IA personnalisé, le système doit être en mesure d'accéder à ce fichier manifest sur un serveur HTTP.</p>

## Installation à partir d'un support IA

Pour effectuer une installation avec initialisation à partir d'une image IA située sur un support, procédez comme suit :

### 1. Téléchargez l'image d'initialisation IA.

Pour télécharger l'image d'initialisation IA, accédez à la page Web suivante :

<http://www.oracle.com/technetwork/server-storage/solaris11/downloads/index.html>

#### ▪ Systèmes SPARC

Pour les systèmes SPARC, téléchargez le fichier IA SPARC .iso.

#### ▪ Systèmes x86

Pour les systèmes x86, téléchargez le fichier IA x86 .iso ou le fichier IA x86 .usb.

### 2. Vérifiez le fichier manifest IA par défaut.

Vous pouvez utiliser le fichier manifest par défaut qui est fourni avec l'image IA ou créer un fichier manifest personnalisé et indiquer l'emplacement de ce fichier lors de l'initialisation du client. Reportez-vous à la section “[Création d'un fichier manifest IA personnalisé](#)” à la page 103.

### 3. Créez un support de démarrage.

#### ▪ Images SPARC et x86 ISO

Copiez le fichier .iso sur un CD ou un DVD.

#### ▪ Images x86 USB

Utilisez `usbcopy(1)` ou un utilitaire similaire pour créer un support de démarrage à l'aide du fichier .usb.

### 4. Effectuez l'initialisation à partir du support.

Initialisez le système à partir du périphérique qui contient l'image d'initialisation.

Reportez-vous aux rubriques “[Initialisation d'un système SPARC à partir d'un support IA](#)” à la page 104 et “[Initialisation d'un système x86 à partir d'un support IA](#)” à la page 105 pour obtenir des instructions sur la manière de spécifier le fichier manifest IA par défaut ou un fichier manifest IA personnalisé et poursuivez l'installation mains libres.

## Création d'un fichier manifest IA personnalisé

Vous pouvez effectuer l'installation du système à l'aide des spécifications d'installation contenues dans le fichier manifest IA de l'image d'initialisation IA. Vous pouvez également créer vos propres spécifications d'installation. Si vous créez un fichier manifest IA personnalisé, enregistrez-le sur un serveur HTTP et indiquez son emplacement lorsque vous initialisez le système à installer.

Si vous téléchargez l'image IA .iso, vous pouvez utiliser les commandes suivantes pour vérifier le fichier manifest IA dans cette image. Dans cet exemple, /tmp est le répertoire dans lequel vous

avez téléchargé l'image IA, et /home/username est le répertoire dans lequel vous voulez copier et modifier le fichier manifest IA. Le fichier manifest IA se trouve dans le fichier auto-installer/default.xml dans l'image.

```
# lofi_dev=$(/usr/sbin/lofiadm -a /tmp/sol-11-ai-sparc.iso)
# /usr/sbin/mount -o ro -F hsfs ${lofi_dev} /mnt
# cp /mnt/auto_install/default.xml /home/username/custom.xml
# /usr/sbin/umount /mnt
# /usr/sbin/lofiadm -d ${lofi_dev}
```

Vérifiez votre copie du fichier manifest par défaut (/home/username/custom.xml dans cet exemple) et décidez si ces spécifications conviennent à cette installation.

Vous pouvez également utiliser le fichier manifest indiqué dans [“Fichier manifest IA par défaut”](#) à la page 37 comme base pour créer un fichier manifest personnalisé.

Pour modifier les spécifications d'installation telles que le disque cible ou d'autres packages à installer, consultez le [Chapitre 4, “Spécification des instructions d'installation”](#). Pour changer les spécifications de configuration telles que le compte d'utilisateur ou le mot de passe root, reportez-vous au [Chapitre 5, “Configuration du système client”](#).

Lorsque vous avez terminé de modifier le fichier manifest IA, copiez le fichier manifest personnalisé sur un serveur HTTP. Notez l'URL du fichier manifest IA personnalisé afin de pouvoir fournir cet URL lorsque vous initialisez le système à installer. Par exemple, l'URL peut être `http://example.com/custom.xml`.

## Initialisation d'un système SPARC à partir d'un support IA

Vous pouvez spécifier le fichier manifest IA par défaut ou un fichier manifest IA personnalisé lorsque vous initialisez le système à partir du support IA.

### Utilisation du fichier manifest IA par défaut

Pour utiliser le fichier manifest IA par défaut contenu dans l'image d'initialisation IA, entrez la commande suivante à l'invite OBP :

```
ok> boot cdrom - install
```

L'installation personnalisée commence à l'aide des spécifications indiquées dans le fichier manifest par défaut.

### Utilisation d'un fichier manifest IA personnalisé

Pour utiliser un fichier manifest IA, entrez la commande suivante à l'invite OBP :

```
ok> boot cdrom - install prompt
```

Le message suivant s'affiche :



Enter the URL for the AI manifest [HTTP, default]:

Entrez l'URL de votre fichier manifest personnalisé. Par exemple, tapez  
`http://example.com/custom.xml` .

L'installation automatisée commence à l'aide des spécifications indiquées dans le fichier manifest par défaut.

## Initialisation d'une image SPARC sans installation

Vous pouvez être amené à effectuer l'initialisation à partir d'un support mais sans effectuer d'installation. Par exemple, vous pouvez être amené à dépanner ou examiner le système.

Pour initialiser l'image IA sans démarrer une installation automatisée, utilisez la commande suivante :

```
ok> boot cdrom
```

Le système démarre, et un écran de connexion s'affiche, mais l'installation ne commence pas.

## Initialisation d'un système x86 à partir d'un support IA

Sur un système x86, choisissez une option d'installation automatisée à partir du menu de GRUB. Selon l'option du menu GRUB ou la commande d'initialisation choisie, l'installation utilise le fichier manifest par défaut sur le support ou un fichier manifest personnalisé que vous avez stocké sur un serveur HTTP.

Les sélections que vous effectuez dans le menu GRUB doivent être similaires à l'exemple suivant :

```
GNU GRUB version 0.97 (639K lower / 2078660K upper memory)
```

```
Oracle Solaris 11 Express Automated Install custom
Oracle Solaris 11 Express Automated Install
Oracle Solaris 11 Express Automated Install custom ttya
Oracle Solaris 11 Express Automated Install custom ttyb
Oracle Solaris 11 Express Automated Install ttya
Oracle Solaris 11 Express Automated Install ttyb
Boot from Hard Disk
```

Use the arrow keys to select which entry is highlighted.  
 Press enter to boot the selected OS, 'e' to edit the  
 commands before booting, or 'c' for a command-line.

## Utilisation du fichier manifest IA par défaut

Pour utiliser le fichier manifest IA par défaut contenu dans l'image d'initialisation IA, utilisez les touches fléchées pour sélectionner l'une des options suivantes :

```
Oracle Solaris 11 Express Automated Install
Oracle Solaris 11 Express Automated Install ttya
Oracle Solaris 11 Express Automated Install ttyb
```

L'option `ttya` envoie la sortie écran au cours de l'installation à la console série `ttya` (COM1).  
L'option `ttyb` envoie la sortie écran au cours de l'installation à la console série `ttyb` (COM2).

L'installation personnalisée commence à l'aide des spécifications indiquées dans le fichier manifest par défaut.

## Utilisation d'un fichier manifest IA personnalisé

Pour utiliser un fichier manifest IA personnalisé, choisissez l'une des options suivantes :

```
Oracle Solaris 11 Express Automated Install custom
Oracle Solaris 11 Express Automated Install custom ttya
Oracle Solaris 11 Express Automated Install custom ttyb
```

Lorsque vous sélectionnez l'une de ces options personnalisées, le message suivant s'affiche :

```
Enter the URL for the AI manifest [HTTP, default]:
```

Entrez l'URL de votre fichier manifest personnalisé. Par exemple, tapez  
`http://example.com/custom.xml`.

L'installation automatisée commence à l'aide des spécifications indiquées dans le fichier manifest par défaut.

## Initialisation d'une image x86 sans installation

Vous pouvez être amené à effectuer l'initialisation à partir d'un support mais sans effectuer d'installation. Par exemple, vous pouvez être amené à dépanner ou examiner le système.

En général, si `install=true` est spécifié dans la ligne de noyau pour l'entrée GRUB que vous utilisez, l'installation démarre automatiquement. Si vous avez l'intention d'initialiser le système x86 sans lancer immédiatement une installation automatisée, vérifiez l'entrée de menu GRUB que vous allez utiliser. Si `install=true` est spécifié dans la ligne de noyau pour cette entrée GRUB, modifiez la ligne en supprimant `install=true`. Ensuite, lorsque vous choisissez cette option, le système s'initialise et un écran de connexion s'affiche, mais l'installation ne démarre pas.

## Affichage des fichiers journaux d'installation

Lorsque l'installation automatisée est terminée, la sortie indique si l'opération a échoué ou réussi.

- En cas d'échec de l'installation, vous pouvez consulter le journal de l'installation sous `/tmp/install_log`.
- Si l'installation a réussi, vous pouvez consulter le journal, disponible sous `/tmp/install_log`, avant la réinitialisation du système, ou `/var/sadm/system/logs/install_log` après la réinitialisation.

# Dépannage des installations automatisées

---

Cette annexe traite de plusieurs pannes possibles et des solutions existantes.

## Échec de l'installation du client

Cette section vous suggère les actions à entreprendre si l'installation d'un client échoue.

### Vérifier les journaux d'installation

Si une installation sur un système client a échoué, vous pouvez trouver le journal sous `/tmp/certains`.

### Vérifier le référentiel IPS

Le client d'installation doit atteindre le référentiel IPS défini dans le fichier manifest pour installer le SE Oracle Solaris. Dans une configuration normale, le serveur DHCP envoie les informations DNS au client. Ces informations DNS sont utilisées pour résoudre le nom du référentiel IPS en adresse IP.

Un échec d'installation peut entraîner l'affichage d'un message d'erreur. Reportez-vous à l'exemple de message d'erreur ci-après :

```
<OM Nov 2 06:56:32> Creating and configuring pkg(5) image area...
  <TRANSFER_MOD Nov 2 06:56:32> pkg cmd: /usr/bin/pkg image-create
-f -F -p example.com=http://pkg.example.com/release /a
  <TRANSFER_MOD_E Nov 2 06:56:35>
  <TRANSFER_MOD_E Nov 2 06:56:35>
  <TRANSFER_MOD_E Nov 2 06:56:35> pkg image-create: The URI
'http://pkg.example.com/release' does not appear to point to a valid pkg server.
  <TRANSFER_MOD_E Nov 2 06:56:35> Please check the server's address and client's
```

```
network configuration.
<TRANSFER_MOD_E Nov  2 06:56:35> Additional details:
<TRANSFER_MOD_E Nov  2 06:56:35>
<TRANSFER_MOD_E Nov  2 06:56:35> Unable to contact valid package server
<TRANSFER_MOD_E Nov  2 06:56:35> Encountered the following error(s):
<TRANSFER_MOD_E Nov  2 06:56:35> Unable to contact any configured publishers.
This is likely a network configuration problem.
<TRANSFER_MOD_E Nov  2 06:56:35> Unable to initialize the pkg image area at /a
...
<AI Nov  2 06:56:36> Automated Installation failed in Transfer module
<AI Nov  2 06:56:36> Transferring the files from the source failed.
Please see previous messages for more details
```

Ce message d'erreur indique que le client n'a pas pu résoudre le nom du référentiel IPS. Sur la machine en panne, essayez d'atteindre le référentiel IPS. Cette sortie indique que le référentiel que ce client d'installation essaie d'atteindre est `http://pkg.example.com/release`.

Vérifiez si le client peut envoyer un ping au référentiel :

```
$ ping pkg.example.com
```

Si la commande `ping` renvoie la sortie suivante, l'erreur peut correspondre à un problème de connectivité :

```
no answer from pkg.example.com
```

Si la commande `ping` renvoie la sortie suivante, l'erreur peut correspondre à un problème lié au service DNS :

```
ping: unknown host pkg.example.com
```

## Vérifier DNS

Vérifiez si le service DNS est configuré sur votre client en vérifiant qu'il existe un fichier non vide `/etc/resolv.conf`.

Si `/etc/resolv.conf` n'existe pas ou est vide, vérifiez que votre serveur DHCP fournit des informations relatives au serveur DNS au client :

```
# /sbin/dhccpinfo DNSserv
```

Si cette commande ne renvoie rien, cela signifie que le serveur DHCP n'est pas configuré pour fournir des informations relatives au serveur DNS au client. Contactez votre administrateur DHCP pour corriger ce problème.

Si un fichier `/etc/resolv.conf` existe et est configuré correctement, vérifiez la présence éventuelle des problèmes suivants et contactez votre administrateur système pour les résoudre :

- Il se peut que le serveur DNS ne parvienne pas à résoudre le nom de votre serveur de référentiel IPS.
- Il n'existe aucune route par défaut permettant d'atteindre le serveur DNS.

## Vérifier les erreurs d'initialisation du client

Consultez les informations supplémentaires suivantes, concernant les erreurs qui se produisent lorsque le système client est en cours d'initialisation.

- [“Erreurs d'initialisation réseau SPARC et causes possibles”](#) à la page 109
- [“Erreurs d'initialisation réseau x86 et causes possibles”](#) à la page 112
- [“Messages d'erreur SPARC et x86”](#) à la page 115

## Erreurs d'initialisation réseau SPARC et causes possibles

Cette section décrit les erreurs ou les problèmes qui peuvent survenir lors de l'initialisation d'un client SPARC sur le réseau et les causes possibles.

- [“Timed out Waiting for BOOTP/DHCP Reply \(Expiration du délai d'attente de la réponse BOOTP/DHCP\)”](#) à la page 109
- [“Boot Load Failed \(Échec du chargement d'initialisation\)”](#) à la page 110
- [“Internal Server Error or WAN Boot Alert \(Erreur de serveur interne ou alerte d'initialisation via une connexion WAN\)”](#) à la page 110
- [“Error Message 403: Forbidden or 404 Not Found \(Message d'erreur 403 : interdit ou 404 introuvable\)”](#) à la page 111
- [“Automated Installer Disabled \(Programme d'installation automatique désactivé\)”](#) à la page 112

### Timed out Waiting for BOOTP/DHCP Reply (Expiration du délai d'attente de la réponse BOOTP/DHCP)

Si un serveur DHCP ne répond pas à une demande d'un client SPARC, les messages suivants s'affichent :

```
...
OpenBoot 4.23.4, 8184 MB memory available, Serial #69329298.
Ethernet address 0:14:4f:21:e1:92, Host ID: 8421e192.
Rebooting with command: boot net:dhcp - install
Boot device: /pci@7c0/pci@0/network@4:dhcp File and args:
1000 Mbps FDX Link up
Timed out waiting for BOOTP/DHCP reply
Timed out waiting for BOOTP/DHCP reply
Timed out waiting for BOOTP/DHCP reply
Timed out waiting for BOOTP/DHCP reply
```

Le message de délai d'attente indique que le client envoie une requête DHCP, mais n'obtient aucune réponse. Cette erreur est probablement due à un problème de configuration DHCP. Vérifiez si le client est correctement configuré sur le serveur DHCP.

## Boot Load Failed (Échec du chargement d'initialisation)

Si le client commence le téléchargement de l'archive `boot_archive` et si cette opération échoue lorsque l'erreur "Boot load failed" se produit, cela indique que les informations DHCP du client ne sont pas correctement configurées.

```
Rebooting with command: boot net:dhcp - install
  Boot device: /pci@7c0/pci@0/network@4:dhcp File and args:
  1000 Mbps FDX Link up
  HTTP: Bad Response: 500 Internal Server Error
  Evaluating:
```

```
  Boot load failed
```

Cette erreur peut se produire si un autre serveur DHCP répond au client. Vérifiez la configuration DHCP pour ce client. Si la configuration semble correcte, déterminez s'il existe un autre serveur DHCP sur le sous-réseau. Si vous exécutez le serveur DHCP Oracle Solaris, vous pouvez exécuter le démon DHCP en mode de débogage au moyen de la commande suivante :

```
# /usr/lib/inet/in.dhcpd -dv
```

## Internal Server Error or WAN Boot Alert (Erreur de serveur interne ou alerte d'initialisation via une connexion WAN)

Une fois que le client a obtenu l'adresse IP et les paramètres initiaux pour commencer à télécharger l'archive d'initialisation, il se peut qu'il ne parvienne pas à localiser ou à télécharger le fichier `boot_archive`.

- Si le fichier `boot_archive` est introuvable, le message d'erreur suivant s'affiche.

```
Rebooting with command: boot net:dhcp - install
  Boot device: /pci@7c0/pci@0/network@4:dhcp File and args:
  1000 Mbps FDX Link up
  <time unavailable> wanboot info: WAN boot messages->console
  <time unavailable> wanboot info: Starting DHCP configuration
  <time unavailable> wanboot info: DHCP configuration succeeded
  <time unavailable> wanboot progress: wanbootfs: Read 366 of 366 kB (100%)
  <time unavailable> wanboot info: wanbootfs: Download complete
  Tue Aug 5 20:46:43 wanboot alert: miniinfo: Request returned code 500
  Tue Aug 5 20:46:44 wanboot alert: Internal Server Error \
  (root filesystem image missing)
```

- Si le client trouve le fichier `boot_archive`, mais ne peut y accéder, l'erreur suivante s'affiche.

```
Rebooting with command: boot net:dhcp - install
  Boot device: /pci@7c0/pci@0/network@4:dhcp File and args:
  1000 Mbps FDX Link up
```

```

<time unavailable> wanboot info: WAN boot messages->console
<time unavailable> wanboot info: Starting DHCP configuration
<time unavailable> wanboot info: DHCP configuration succeeded
<time unavailable> wanboot progress: wanbootfs: Read 366 of 366 kB (100%)
<time unavailable> wanboot info: wanbootfs: Download complete
Tue Aug 5 20:53:02 wanboot alert: miniroot: Request returned code 403
Tue Aug 5 20:53:03 wanboot alert: Forbidden

```

Pour ces deux problèmes, corrigez le fichier `boot_archive` configuré pour ce client. Vérifiez le nom du chemin et les autorisations du fichier `boot_archive` sous `$Image/boot/boot_archive`.

## Error Message 403: Forbidden or 404 Not Found (Message d'erreur 403 : interdit ou 404 introuvable)

Ces messages, "ERROR 403: Forbidden" et "ERROR 404: Not Found", s'affichent si le client IA parvient à télécharger le fichier `boot_archive` et initialise le noyau Oracle Solaris, mais ne peut récupérer l'une des archives d'image. Un message d'erreur s'affiche, indiquant le fichier qui pose problème. Par exemple, dans la sortie ci-dessous, le fichier `solaris.zlib` n'existe pas ou n'est pas accessible à l'emplacement spécifié.

```

Rebooting with command: boot net:dhcp - install
Boot device: /pci@7c0/pci@0/network@4:dhcp File and args:
1000 Mbps FDX Link up
1000 Mbps FDX Link up
<time unavailable> wanboot info: WAN boot messages->console
<time unavailable> wanboot info: Starting DHCP configuration
<time unavailable> wanboot info: DHCP configuration succeeded
<time unavailable> wanboot progress: wanbootfs: Read 366 of 366 kB (100%)
<time unavailable> wanboot info: wanbootfs: Download complete
Tue Aug 5 21:43:37 wanboot progress: miniroot: Read 165251 of 165251 kB (100%)
Tue Aug 5 21:43:38 wanboot info: miniroot: Download complete
SunOS Release 5.11 Version snv_151 64-bit
...
Hostname: solaris
Remounting root read/write
Probing for device nodes ...
Preparing automated install image for use
Downloading solaris.zlib archive
--15:40:37-- http://10.6.35.226:5555//export/home/images/ai_sparc_111/solaris.zlib
=> '/tmp/solaris.zlib'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 403 Forbidden
15:40:37 ERROR 403: Forbidden.

FAILED
Requesting System Maintenance Mode
(See /lib/svc/share/README for more information.)
Console login service(s) cannot run

```

Ce problème peut être dû à l'une des conditions suivantes.

- Le chemin de l'image défini pour l'initialisation via une connexion WAN est incorrect.
- Le chemin de l'image n'existe pas ou est incomplet.
- L'accès est refusé en raison de problèmes d'autorisation.

Vérifiez votre configuration DHCP ou le contenu du répertoire cible que vous avez spécifié lorsque vous avez exécuté `installadm create-service`. Vérifiez la configuration de l'initialisation via une connexion WAN.

## Automated Installer Disabled (Programme d'installation automatique désactivé)

Pour l'installation du SE Oracle Solaris sur votre système client, vous devez inclure l'argument `install` lors de l'initialisation de la manière suivante afin de lancer une installation.

```
ok> boot net:dhcp - install
```

Si vous avez effectué l'initialisation sans l'argument d'initialisation `install`, le client SPARC s'initialise dans l'image d'initialisation du programme d'installation automatisée, mais l'installation ne démarre pas. Le message suivant s'affiche :

```
Auto-installer disabled. Enable the auto-installer service
by running the following command:
svcadm enable svc:/application/auto-installer:default
```

Pour lancer une installation automatisée, vous pouvez vous connecter et activer le service d'installation comme indiqué dans le message. Vous pouvez également redémarrer votre système à l'aide de la commande ci-dessus et de l'argument `install`.

## Erreurs d'initialisation réseau x86 et causes possibles

Cette section décrit les erreurs ou les problèmes qui peuvent survenir lors de l'initialisation d'un client x86 sur le réseau et les causes possibles :

- “No DHCP or ProxyDHCP Offers Were Received (Aucune offre DHCP ou ProxyDHCP n'a été reçue)” à la page 112
- “TFTP Error or System Hangs After GATEWAY Message (Erreur TFTP ou interruption du système suite à un message de passerelle)” à la page 113
- “System Hangs After GRUB Menu Entry is Selected (Interruption du système après sélection de l'entrée de menu GRUB)” à la page 113
- “HTTP Request Sent Results in 403 Forbidden or 404 Not Found (La requête HTTP envoyée entraîne les erreurs "403 Forbidden" ou "404 Not Found")” à la page 114
- “Automated Installer Disabled (Programme d'installation automatique désactivé)” à la page 114

## No DHCP or ProxyDHCP Offers Were Received (Aucune offre DHCP ou ProxyDHCP n'a été reçue)

Si un serveur DHCP ne répond pas à une requête d'un client x86, les messages suivants s'affichent :



```
Intel(R) Boot Agent PXE Base Code (PXE-2.1 build 0.86)
Copyright(C) 1997-2007, Intel Corporation
```

```
CLIENT MAC ADDR 00 14 4F 29 04 12 GUID FF2000008 FFFF FFFF FFFF 7BDA264F1400
DHCP..... No DHCP or ProxyDHCP offers were received
PXE-MOF: Exiting Intel Boot Agent
```

Le message de délai d'attente indique que le client envoie une requête DHCP, mais n'obtient aucune réponse. Le problème est probablement dû à une erreur de configuration DHCP. Vérifiez si le client est correctement configuré sur le serveur DHCP.

## TFTP Error or System Hangs After GATEWAY Message (Erreur TFTP ou interruption du système suite à un message de passerelle)

Le serveur DHCP fournit une adresse IP et un emplacement du programme d'initialisation initiale dans la réponse DHCP.

- Si le programme d'initialisation n'existe pas, l'initialisation du client IA ne peut pas continuer. Le message suivant s'affiche :

```
Intel(R) Boot Agent PXE Base Code (PXE-2.1 build 0.86)
Copyright(C) 1997-2007, Intel Corporation
```

```
CLIENT MAC ADDR 00 14 4F 29 04 12 GUID FF2000008 FFFF FFFF FFFF 7BDA264F1400
CLIENT IP: 10.6.68.29 MASK: 255.255.255.0 DHCP IP: 10.6.68.49
GATEWAY: 10.6.68.1
TFTP.
PXE-T02: Access Violation
PXE-E3C: TFTP Error - Access violation
PXE-MOF: Exiting Intel Boot Agent
```

- Si le programme d'initialisation existe, mais qu'il s'agit d'un programme incorrect, le client IA s'arrête après l'affichage du message suivant :

```
Intel(R) Boot Agent PXE Base Code (PXE-2.1 build 0.86)
Copyright(C) 1997-2007, Intel Corporation
```

```
CLIENT MAC ADDR 00 14 4F 29 04 12 GUID FF2000008 FFFF FFFF FFFF 7BDA264F1400
CLIENT IP: 10.6.68.29 MASK: 255.255.255.0 DHCP IP: 10.6.68.49
GATEWAY: 10.6.68.1
```

## System Hangs After GRUB Menu Entry is Selected (Interruption du système après sélection de l'entrée de menu GRUB)

Si le client est en mesure d'effectuer l'initialisation initiale alors que le noyau n'est pas initialisé, le système se bloque après la sélection de l'entrée du menu GRUB par l'utilisateur.

Sur le serveur d'installation, vérifiez que le fichier menu.lst correspondant à ce client fait référence à une archive d'initialisation valide. Le répertoire d'initialisation de l'image sur le serveur doit être monté en boucle sous le répertoire /tftpboot comme indiqué dans cet exemple de df -k :

```
/export/home/images/osol-1003-ai-x86/boot \
60450439 21678071 38772368 36% /tftpboot/I86PC.Solaris-12
```

I86PC.OpenSolaris-12 est un exemple. Le numéro peut varier d'un système à l'autre.

Si vous connaissez le nom du répertoire cible que vous avez spécifié dans la commande `installadm create-service`, vous pouvez utiliser ces informations pour déterminer si ce répertoire cible est monté. Vérifiez également si vous pouvez accéder au fichier `/tftpboot/I86PC.Solaris-12/boot_archive`.

## HTTP Request Sent Results in 403 Forbidden or 404 Not Found (La requête HTTP envoyée entraîne les erreurs "403 Forbidden" ou "404 Not Found")

Sur le serveur d'installation, si l'un des programmes d'installation n'est pas accessible ou n'existe pas à l'emplacement spécifié dans le fichier `menu.lst` sous `/tftpboot`, le client peut alors s'initialiser, mais n'est pas en mesure de télécharger ce fichier. Un message d'erreur s'affiche, indiquant le fichier qui pose problème. Par exemple, dans la sortie ci-dessous, le fichier `solaris.zlib` n'existe pas à l'emplacement spécifié.

```
SunOS Release 5.11 Version snv_151 64-bit
...
Hostname: solaris
Remounting root read/write
Probing for device nodes ...
Preparing automated install image for use
Downloading solaris.zlib archive
--15:40:37-- http://10.6.35.226:5555//export/home/images/ai_x86_111/solaris.zlib
=> '/tmp/solaris.zlib'
Connecting to 10.6.35.226:5555... connected.
HTTP request sent, awaiting response... 403 Forbidden
15:40:37 ERROR 403: Forbidden.

FAILED
Requesting System Maintenance Mode
(See /lib/svc/share/README for more information.)
Console login service(s) cannot run
```

Vérifiez le contenu du répertoire cible que vous avez spécifié lors de l'exécution de la commande `installadm create-service`.

## Automated Installer Disabled (Programme d'installation automatique désactivé)

Lorsque vous installez le SE Oracle Solaris sur un système client x86, pour les installations qui s'initialisent sur le réseau, vous devez sélectionner la seconde entrée du menu d'initialisation GRUB pour lancer une installation automatisée. En règle générale, les entrées du menu s'affichent comme suit :

```
Oracle Solaris 11 Express boot image
Oracle Solaris 11 Express Automated Install
```

Si vous avez sélectionné la première entrée du menu GRUB ou autorisé l'expiration de l'invite, le système initialise dans l'image d'initialisation d'installation automatisée, mais l'installation ne démarre pas. Le message suivant s'affiche :

Auto-installer disabled. Enable the auto-installer service by running the following command:  
`svcadm enable svc:/application/auto-installer:default`

Pour lancer une installation automatisée, vous pouvez vous connecter et activer le service d'installation comme indiqué dans le message. Vous pouvez également redémarrer le système et sélectionner la deuxième entrée de menu.

## Messages d'erreur SPARC et x86

Les erreurs suivantes sont communes aux deux installations SPARC et x86.

- “Automated Installation Failed (Échec de l'installation automatisée)” à la page 115
- “Unable to Contact Valid Package Server (Impossible de contacter le serveur de package valide)” à la page 115
- “Install Log Reports Missing Package (Le journal d'installation signale un package manquant)” à la page 116

### Automated Installation Failed (Échec de l'installation automatisée)

En cas d'erreur lors de l'installation, le message suivant s'affiche :

```
Automated Installation failed. Please refer to /tmp/install_log file for
  details
Apr  9 14:28:09 solaris svc.startd[7]: application/auto-installer:default
  failed fatally: transitioned to maintenance (see 'svcs -xv' for details)
```

### Unable to Contact Valid Package Server (Impossible de contacter le serveur de package valide)

En cas d'erreur lors de l'installation, consultez le fichier journal /tmp/install\_log. Si le client IA n'a pas pu accéder au serveur de package pour l'installation des packages, le système peut afficher les messages d'erreur suivants :

```
installation will be performed from http://pkg.oracle.com/ (solaris)
  list of packages to be installed is:
      entire
      babel_install
pkg: The URL 'http://pkg.oracle.com/' does not appear to point to a
  valid pkg server.
Please check the server's address and client's network configuration.
Additional details:

Unable to contact valid package server: http://pkg.oracle.com/
Encountered the following error(s):
Transport errors encountered when trying to contact depot server.
Reported the following errors:
Could not retrieve versions from 'solaris'
URL error, reason: (8, 'node name or service name not known')
Unable to initialize the pkg image area at /a
```

Le problème est peut-être dû à l'une des causes suivantes :

- Le serveur pkg\_server est en panne. Essayez d'accéder au pkg\_server et vérifiez le statut.
- Si vous utilisez DNS, vérifiez que le client IA est configuré avec DNS : Voir [“Vérifier DNS” à la page 108](#).

## Install Log Reports Missing Package (Le journal d'installation signale un package manquant)

Si l'un des packages spécifiés dans le fichier manifest IA ne peut pas être localisé dans les référentiels IPS, le programme d'installation se bloque avant d'installer tout package sur le disque. Dans l'exemple suivant, le programme d'installation n'a pas pu trouver le package SUNWTestPkg dans le référentiel IPS. Il est possible que le message d'erreur suivant s'affiche dans le fichier install\_log :

```
<AI Feb 12 20:15:40> installation will be performed from
http://pkg.oracle.com/solaris/release (solaris)
<AI Feb 12 20:15:40> list of packages to be installed is:
<AI Feb 12 20:15:40> entire
    babel_install
<AI Feb 12 20:15:40> SUNWTestPkg
<OM Feb 12 20:15:40> Set zfs root pool device
<OM Feb 12 20:15:40> creating zpool
<OM Feb 12 20:15:43> /usr/sbin/zfs get -Hp -o value available rpool
<OM Feb 12 20:15:43> Creating swap and dump on ZFS volumes
<OM Feb 12 20:16:00> TI process completed successfully
<OM Feb 12 20:16:00> Transfer process initiated
<TRANSFERMOD Feb 12 20:16:59> IPS initialization finished
<TRANSFER_MOD Feb 12 20:52:33> pkg missing
<TRANSFERMOD Feb 12 20:52:33> IPS transfer failed
<TRANSFERMOD Feb 12 20:52:33> IPS transfer finished
<OM Feb 12 20:52:33> Transfer failed with error 15
<AI Feb 12 20:52:42> om_perform_install failed with error 114
<AI Feb 12 20:52:42> Auto install failed
```

Vérifiez si le package en question est un package valide. Si ce package est disponible sur un autre référentiel IPS, ajoutez ce référentiel IPS au fichier manifest IA à l'aide de la balise <source>.

## Run Automated Installations in Debug Verbose Mode (Exécuter les installations automatisées en mode de débogage détaillé)

Vous pouvez exécuter le programme d'installation en mode de débogage détaillé pour capturer davantage d'informations sur l'installation dans le fichier install\_log.

## Boot SPARC Systems in Debug Mode (Initialiser les systèmes SPARC en mode de débogage)

Pour initialiser les systèmes SPARC en mode de débogage, initialisez le système à l'aide de l'argument d'initialisation `install_debug`.

Pour les installations réseau :

```
ok> boot net:dhcp - install install_debug
```

Pour les installations basées sur un support :

```
ok> boot cdrom - install install_debug
```

## Boot x86 Systems in Debug Mode (Initialiser les systèmes x86 en mode de débogage)

Pour initialiser les systèmes x86 en mode de débogage, modifiez le menu GRUB et ajoutez la propriété d'initialisation `install_debug=enable` à la ligne du noyau de l'entrée à initialiser.

```
kernel$ ... -B install_media=...,install_debug=enable
```

Pour les systèmes x86, vous pouvez prédéfinir le mode de débogage sur le serveur IA en utilisant l'option `-b <property=value>` avec les commandes `installadm create-service` et `installadm create-client`. Cette option prédéfini cette propriété d'initialisation dans le menu GRUB pour ce service ou ce client.

```
# installadm create-client -e MAC_address -n service_name \
-b install_debug=enable
```

## Boot the Install Environment Without Starting an Installation (Initialiser l'environnement d'installation sans démarrer d'installation)

Pour initialiser l'environnement d'installation sans lancer une installation, effectuez l'initialisation sans spécifier l'indicateur `install` en tant qu'argument d'initialisation.

- Pour un système SPARC, effectuez l'initialisation à l'aide de l'une des commandes suivantes :

```
ok> boot net:dhcp
ok> boot cdrom
```

Étant donné que vous n'avez pas inclus l'argument `install` dans la commande, l'installation automatisée ne démarre pas.

- Pour un système x86, effectuez l'initialisation avec l'une des options suivantes.
  - Pour les installations x86 initialisées sur le réseau, le menu GRUB s'affiche comme suit.

```
Oracle Solaris 11 Express boot image
Oracle Solaris 11 Express Automated Install
```

L'entrée de menu GRUB par défaut, "Oracle Solaris 11 Express boot image" (image d'initialisation d'Oracle Solaris 11 Express), initialise l'image sans démarrer une installation automatisée "mains libres".

---

**Remarque** – Pour vous assurer que le système est initialisé sans démarrer l'installation, vérifiez que la propriété d'initialisation `install=true` de l'entrée que vous souhaitez initialiser n'est pas spécifiée dans la ligne de noyau de cette dernière.

---

- Si vous initialisez un système x86 à partir d'un support, modifiez le menu GRUB et supprimer la propriété d'initialisation `install=true` de la ligne de noyau à partir de l'entrée que vous souhaitez initialiser.

---

**Remarque** – En général, pour les installations x86, si la propriété d'initialisation `install=true` est spécifiée dans la ligne de noyau de l'entrée GRUB à partir de laquelle vous effectuez l'initialisation, l'installation démarre automatiquement. Si vous avez l'intention d'initialiser votre système x86 sans lancer une installation automatisée, vérifiez que l'entrée d'initialisation GRUB ne spécifie pas la propriété d'initialisation `install=true`. Si la propriété est spécifiée, modifiez la ligne de noyau de l'entrée d'initialisation et supprimez la propriété.

---

## Start Installation After Booting Without Initiating an Installation (Démarrer l'installation après l'initialisation sans lancer une installation)

Si vous avez sélectionné une option d'initialisation qui ne lance pas d'installation, utilisez la commande `svcadm` pour activer le service d'installation automatisée.

```
# svcadm enable auto-installer:default
```

# Commandes d'administration de l'installation automatisée

---

Cette annexe présente la page de manuel `installadm(1M)`.

## Page de manuel `installadm(1M)`

System Administration Commands `installadm(1M)`

### NAME

`installadm` - Manages automated installations on a network

### SYNOPSIS

`/usr/sbin/installadm`

```
installadm create-service [-b <property>=<value>,...]
[-f <bootfile>] [-n <svcname>] [-i <dhcp_ip_start>
-c <count_of_ipaddr>] [-s <srcimage>] <targetdir>
```

```
installadm delete-service [-x] <svcname>
```

```
installadm list [-n <svcname>] [-c] [-m]
```

```
installadm enable <svcname>
```

```
installadm disable <svcname>
```

```
installadm add-manifest -m <manifest> -n <svcname>
[-c <criteria=value|range> ... | -C <criteria.xml>]
```

```
installadm delete-manifest -m <manifest> -n <svcname>
```

```
installadm set-criteria -m <manifest> -n <svcname>
-a|-c <criteria=value|range> ... | -C <criteria.xml>
```

```
installadm create-client [-b <property>=<value>,...]
[-t <imagepath>] -e <macaddr> -n <svcname>
```

```
installadm delete-client <macaddr>
```

```
installadm help [<subcommand>]
```

#### DESCRIPTION

The automated installer (AI) is used to automate the installation of the Oracle Solaris OS on one or more SPARC and x86 systems over a network. The installations can differ in architecture, packages installed, disk capacity, and other parameters.

The minimal configuration necessary to use the automated installer is to have one system as the server and one client on which to install. On the server, you set up an installation service, which is associated with manifests, or specifications, for specific x86 installations or SPARC installations.

Manifests can include information such as a target device, additional packages, partition information, and other parameters. When the client boots, a search is initiated for a manifest that matches the client's machine specifications. When a matching manifest is found, the client is installed with the Oracle Solaris release according to the specifications in the manifest file.

Use the `installadm create-service` command to set up an install server and create an install service.

An install service is a network entity that specifies the parameters for a particular type of installation. These specifications are defined in XML manifest files.

The automated installer uses AI ISO images to create the install services. An AI ISO image is a collection of software in a single file. This image is unpacked when an install service is created and used to create a net image that enables client installations.

Once an install server and an installation service are set up, you can install the Oracle Solaris release to a client on the network, per the default specifications in the install service, by booting the client system.

If you want to perform various types of installations in a network, you can create and manage additional install services tailored for each type of installation by using the `installadm create-service` command. For example, you can set up an install service that installs the Oracle Solaris OS to x86 clients and a service that installs the Oracle Solaris OS to SPARC clients.

If you have clients with varying machine specifications, you can manually create or modify manifests so that the manifests cover those specific machine specifications. Then, you can use the `installadm add-manifest` command to add your new manifests to an install service. You can



also use `add-manifest` to specify criteria to be used in determining which manifest should be selected for an installation. You can change criteria already associated with a manifest with the `installadm set-criteria` command.

If you want a specific client to use a specific install service, you can associate a service to a client by using the `installadm create-client` command.

The `installadm` utility can be used to accomplish the following tasks:

- Set up install services
- Set up installation images
- Set up or remove clients
- Add or delete manifests
- Specify or modify criteria for a manifest
- Enable or disable install services
- List install services
- List manifests for an install service

#### SUBCOMMANDS AND OPTIONS

The `installadm` command has the subcommands and options listed below. Also, see `EXAMPLES` below.

`installadm`

Displays command usage.

```
installadm create-service [-b <property>=<value>,...]
[-f <bootfile>] [-n <svcname>] [-i <dhcp_ip_start>
-c <count_of_ipaddr>] [-s <srcimage>] <targetdir>
```

Creates an install service.

The command provides the following functionality:

- Takes an AI ISO image (`<srcimage>`), unpacks it, and sets up a net image in a target directory (`<targetdir>`). The net image enables client installations.
- Creates an install service and makes it available on the network.

Note the following specifications:

- For SPARC install services, the first install service created on an install server is the service that will be used for all client installations that use the install server. If you want a client to use a different install service from this install server, you must use the `installadm create-client` command to create a client-specific configuration.
- By default, both a net image and an install service are created.
- If an existing install service name is provided, that existing service is used.
- If the `-s` option is not used, and the `<targetdir>` contains a valid net image, then a new install service is created with the existing net image.

- If the `-i` option and the `-c` option are used, and a DHCP server is not yet configured, a DHCP server is configured.
  - If an already-configured DHCP server exists, that DHCP server is updated.
  - If DHCP is running on a remote system, updates can happen through secure shell. User must provide authentication.
- `-b <property>=<value>,...`  
For x86 clients only, sets a property value in the service-specific `menu.lst` file in `/tftpboot`. Use this option to set boot properties that are specific to this service. This option can accept multiple `property=value` pairs.
- `-f <bootfile>`  
Uses this boot file for the install service. If boot file doesn't exist, it is created. If this option is not specified, a boot file is created with a default name.
- `-n <svcname>`  
Uses this install service name instead of default service name.
- Note: If the `-n` option is not used, a unique name for the install service is automatically assigned using the format, `"_install_service_<port_number>"`.
- For example, if the port number that `installadm` selected for an install service is `46510`, and the `-n` option was not used to provide a custom name for the install service, then the install service name will be `"_install_service_46510."`
- `-i <dhcp_ip_start>`  
Sets up a new DHCP server. The IP addresses, starting from `dhcp_address_start`, are set up.
- `-c <count_of_ipaddr>`  
Sets up a total number of IP addresses in the DHCP table equal to the value of the `count_of_ipaddr`. The first IP address is the value of `dhcp_ip_start` that is provided by the `-i` option.
- `-s <srcimage>`  
Specifies location of AI ISO image to use for setting up the install service.

`<targetdir>`

Required: Specifies location to set up net image.

`installadm delete-service [-x] <svcname>`

Deletes an install service. Accomplishes the

following:

- Removes install service from the network.
- Stops the web server that is running for this install service.
- Removes the manifest files and web server configuration for this install service.

- x Deletes the install service and also removes the associated target net image.

<svcname>

Required: Specifies the install service name.

installadm list [-n <svcname>] [-c] [-m]

Lists all enabled install services on a server.

-c

Lists the clients of the install services on a local server.

-m

Lists the manifests associated with the install services on a local server.

-n <svcname>

Lists information about the specific install service on a local server. Or, if the -c option is specified, lists the client information associated with the specified install service. Or, if the -m option is specified, lists the manifests associated with the specified install service.

installadm enable <svcname>

Enables a specified install service. Also, enables the web server associated with the service.

<svcname>

Required: Specifies the name of the install service to be enabled.

installadm disable <svcname>

Disables a specified install service. Also, disables the web server associated with the service.

<svcname>

Required: Specifies the name of the install service to be disabled.

installadm add-manifest -m <manifest> -n <svcname>  
[-c <criteria=value|range> ... | -C <criteria.xml>]

Associates manifests with a specific install service, thus making the manifests available on the network, independently from creating a service. When publishing a non-default manifest, it is required to associate criteria either via criteria entered on the command line (-c) or via a criteria XML file (-C).

- m <manifest>  
Required: Specifies the path name of the manifest to add.
- n <svcname>  
Required: Specifies the name of the install service this manifest is to be associated with.
- c <-c <criteria=value|range> ...>  
Optional: Specifies criteria to be associated with the added non-default manifest. When publishing a default manifest, criteria must not be specified. When publishing a non-default manifest, criteria must be specified.
- C <criteria.xml>  
Optional: Specifies the path name of a criteria XML file containing criteria to be associated with the added non-default manifest. When publishing a default manifest, criteria must not be specified. When publishing a non-default manifest, criteria must be specified.

installadm delete-manifest -m <manifest> -n <svcname>

Deletes a manifest that was published with a specific install service.

- m <manifest>  
Required: Specifies the name of an AI manifest as output by installadm list with -n option.
- n <svcname>  
Required: Specifies the name of the install service this manifest is associated with.

installadm set-criteria -m <manifest> -n <svcname>  
-a|-c <criteria=value|range> ... | -C <criteria.xml>

Updates criteria of an already published manifest. Criteria can be specified via the command line or via a criteria xml file. Criteria must be specified with one of the mutually exclusive options, -a, -c, or -C.

- m <manifest>  
Required: Specifies the name of a manifest.
- n <svcname>  
Required: Specifies the name of the install service this manifest is associated with.

- c <-c <criteria=value|range> ...>  
Optional: Specifies criteria to replace all existing criteria for the manifest.
- a <-a <criteria=value|range> ...>  
Optional: Specifies criteria to be appended to the existing criteria for the manifest. If the criteria specified already exists, the value/range of that criteria is replaced by the specified value/range.
- C <criteria.xml>  
Optional: Specifies the path name of a criteria XML file containing criteria to replace all existing criteria for the manifest.

```
installadm create-client [-b <property>=<value>,...]
[-t <imagepath>] -e <macaddr> -n <svcname>
```

Accomplishes optional setup tasks for a specified client, in order to provide custom client settings that vary from the default settings used by the `installadm create-service` command. Enables user to specify a non-default service name and image path for a client:

- Specifies installation service for that client.
- Sets up DHCP macro, if it doesn't exist.

- b <property>=<value>,...  
For x86 clients only, sets a property value in the client-specific `menu.lst` file in `/tftpboot`. Use this option to set boot properties that are specific to this client. This option can accept multiple `property=value` pairs.
- e <macaddr>  
Required: Specifies a MAC address for the client.
- n <svcname>  
Required: Specifies the install service for client installation.
- t <imagepath>  
Specifies the path of the net image to be used with automated installer.

```
installadm delete-client <macaddr>
```

Deletes an existing client's specific service information that was previously set up using the `installadm create-client` command.

```
<macaddr>
```

Required: Specifies a MAC address for the client.

```
installadm help [<subcommand>]
```

Displays the syntax for the `installadm` utility.

<subcommand>

If subcommand is provided, the command provides the syntax for that subcommand.

## CRITERIA FILES

A criteria XML file allows you to specify criteria for a manifest by passing the file to the `add-manifest` or `set-criteria` commands. Criteria can be specified as a value or a range, by using the following tags.

For a criterion with a specific value:

```
<ai_criteria_manifest>
  <ai_criteria name=XXXX>
    <value>yyyy</value>
  </ai_criteria>
</ai_criteria_manifest>
```

where XXXX is the name of the criterion (e.g. MAC, IPV4, MEM, or ARCH) and yyyy is the value of the criterion.

For a criterion with a range:

```
<ai_criteria_manifest>
  <ai_criteria name=XXXX>
    <range>
      yyyy1
      yyyy2
    </range>
  </ai_criteria>
</ai_criteria_manifest>
```

where XXXX is the name of the criterion (e.g. MAC, IPV4, or MEM) and yyyy1 and yyyy2 are the lower and upper bounds of the range.

Multiple criteria may be specified in the file between the `<ai_criteria_manifest>` and `</ai_criteria_manifest>` tags.

## EXAMPLES

Example 1: Set up an install server and an install service for the first time. The command includes a starting IP address and total count of IP addresses, in order to configure the DHCP server.

```
Example% # installadm create-service -n 0906x86 \
-i 10.6.68.201 -c 5 -s \
/export/aiimages/osol-0906-ai-x86.iso \
/export/aiserver/osol-0906-ai-x86
```

In this example, the terminal displays the progress as follows:

Setting up the target image at

```

/export/aiserver/osol-0906-ai-x86 ...
Registering the service 0906x86._OSInstall._tcp.local
Creating DHCP Server
Created DHCP configuration file.
Created dhcptab.
Added "Locale" macro to dhcptab.
Added server macro to dhcptab - line1-x4100.
DHCP server started.
Added network macro to dhcptab - 10.0.0.0.
Created network table.
copying boot file to
/tftpboot/pxegrub.I86PC.Solaris-1
Service discovery fallback mechanism set up

```

The AI ISO image is at  
 /export/aiimages/osol-0906-ai-x86.iso. The command  
 sets up a net image and an install service that is  
 based on the AI ISO image.

The installation net image is created in the  
 /export/aiserver/osol-0906-ai-x86 target directory.  
 This net image enables client installations.

The progress display shows that the install service,  
 named 0906x86, is created.

A boot file is created, also named 0906x86,  
 under /tftpboot. The client will get this file name  
 through DHCP. The command also creates a link from  
 the net image at /export/aiserver/osol-0906-ai-x86  
 to a web server that is running on port 5555.

The DHCP server and macro is created. The starting  
 IP address is 0.6.68.201. Five IP addresses are  
 allocated for clients. The command results  
 identify the macro as dhcp\_macro\_0906x86.

Example 2: Use the following sample command to set  
 up a client that references a specific install service  
 and a specific net image location.

The install service and net image should already exist.

```

# installadm create-client -b "console=ttya" \
-e 0:e0:81:5d:bf:e0 -t \
/export/aiserver/osol-0906-ai-x86 -n 0906x86

```

In this example, the terminal displays the  
 following output:

```

---
Setting up X86 client...
Service discovery fallback mechanism set up

```

```

Detected that DHCP is not set up on this server.
If not already configured, please create a DHCP macro
named 0100E0815DBFE0 with:
Boot server IP (BootSrvA) : 10.6.68.29
Boot file      (BootFile)  : 0100E0815DBFE0

```

If you are running the Solaris DHCP Server, use the following command to add the DHCP macro, 0100E0815DBFE0:

```
/usr/sbin/dhtadm -g -A -m 0100E0815DBFE0 -d \
:BootSrvA=10.6.68.29:BootFile=0100E0815DBFE0:\
GrubMenu=menu.lst.0100E0815DBFE0:
```

Note: Be sure to assign client IP address(es) if needed (e.g., if running the Solaris DHCP Server, run pntadm(1M)).  
----

In this example, the command creates a client-specific setup for the system with MAC address of 0:e0:81:5d:bf:e0.

This client will use the install service previously set up, named 0906x86, and the net image at /export/aiserver/osol-0906-ai-x86.

Using the -b option, the command sets the console value, <console=ttya>, in the client-specific menu.lst file in /tftpboot.

As shown above, this command outputs the name of the client-specific macro, 0100E0815DBFE0, and its values that need to be added to the DHCP server. If you have a Sun DHCP server, create the above macro on your DHCP server by running the dhtadm command from the output on your DHCP server.

The pntadm(1M) command may need to be called if you set up a Sun DHCP server and client IP addresses need to be assigned. See the pntadm(1M) manpage for more information.

On systems which support graphic interfaces, the DHCP Manager may be used instead of the dhtadm or pntadm commands. See the dhcprmgr(1M) manpage for more information.

Example 3: Use the following sample command to replace the default manifest for an existing install service, service\_092910, with a custom manifest, my\_manifest.xml. The manifest contains a name attribute, name="default", which designates it as the default manifest.

```
# installadm add-manifest -m my_manifest.xml \
-n service_092910
```

Example 4: Use the following sample command to list the install services on a local server:

```
# installadm list
```

In this example, the terminal displays the following output:

Service Name	Status	Arch	Port	Image Path
-----	-----	----	----	-----
svc-2008-11	off	x86	45602	/export/server/osol-0811-ai-x86



```
svc-2009-06 on          x86  45601 /export/server/osol-0906-ai-x86
svc-bld-127 on         x86  45603 /export/server/osol-b127-ai-x86
```

Example 5: Use the following sample command to list the clients for a specific install service of a local server:

```
# installadm list -c -n svc-2009-06
```

In this example, the terminal displays the following output:

```
Service Name Client Address      Arch  Image Path
-----
svc-2009-06  01:C2:52:E6:4B:E1 x86   /export/server/osol-0906-ai-x86
```

Example 6: Use the following sample command to list the manifests associated with a specific install service on a local server:

```
# installadm list -m -n svc-2009-06
```

In this example, the terminal displays the following output:

```
Manifest          Criteria
-----
devpublisher.xml arch      = i86pc
                  ipv4     = 010.000.002.015
                  mac      = 01:C4:51:E6:4B:E6 - 01:C4:51:E6:4B:E9
                  mem      = 2048 MB
```

Example 7: Use the following sample command to add manifest1 to svc1 with a criteria of MAC address equaling "aa:bb:cc:dd:ee:ff":

```
# installadm add-manifest -m manifest1 -n svc1
-c MAC="aa:bb:cc:dd:ee:ff"
```

Example 8: Use the following sample command to add manifest2 to svc1 with a criteria of an IPv4 range between 10.0.2.100 to 10.0.2.199:

```
# installadm add-manifest -m manifest2 -n svc1
-c IPV4="10.0.2.100-10.0.2.199"
```

Example 9: Use the following sample command to add manifest3 to svc1 with a criteria of 2048MB memory or greater and an architecture of i86pc:

```
# installadm add-manifest -m manifest3 -n svc1
-c MEM="2048-unbounded" -c ARCH=i86pc
```

Example 10: Use the following sample command to append to the criteria of manifest2 of svc1, a criterion of 4096MB memory or greater:

```
# installadm set-criteria -m manifest2 -n svc1
-a MEM="4096-unbounded"
```

Example 11: Use the following sample command to replace the criteria of manifest2 of svc1 with the criteria specified by

the file, /tmp/criteria.xml:

```
# installadm set-criteria -m manifest2 -n svc1  
-C /tmp/criteria.xml
```

See the CRITERIA FILES section for more information on the contents of the criteria xml file.

#### ATTRIBUTES

See attributes(5) for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	install/installadm
Interface Stability	None / Under Development

#### SEE ALSO

pntadm(1M), dhcpmgr(1M), attributes(5)

Oracle Solaris Automated Installer Guide on  
<http://docs.sun.com/>

Getting Started with Oracle Solaris on  
<http://docs.sun.com/>

Last Changed October 12, 2010

# Migration de JumpStart vers le programme d'installation automatisée

---

Cette annexe fournit des informations relatives à la migration de Solaris Custom JumpStart vers le programme d'installation automatisée Oracle Solaris.

## Comparaison de JumpStart et du programme d'installation automatisée

Solaris Custom JumpStart et le programme d'installation automatisée (IA) Oracle Solaris permettent l'installation en mode mains libres de plusieurs systèmes sur un réseau. Les clients sont initialisés sur le réseau, et une fois le client initialisé, le programme d'installation prend le relais.

JumpStart installe le système d'exploitation Oracle Solaris 10 et les versions antérieures du système d'exploitation Oracle Solaris. Le programme d'installation automatisée installe le système d'exploitation Oracle Solaris 11 Express.

JumpStart et le programme d'installation automatisée partagent les caractéristiques suivantes :

- Ils permettent l'installation réseau en mode mains libres de plusieurs clients en stockant les configurations système sur un serveur d'installation.
- Ils permettent d'effectuer différents types d'installations sur différents clients en une installation automatisée.
- Installation des clients x86 et SPARC

TABLEAU C-1 Comparaison des tâches JumpStart et du programme d'installation automatisée

Tâche	JumpStart	Programme d'installation automatisée
Définissez un serveur d'installation.	Utilisez la commande <code>setup_install_server(1M)</code> .	Utilisez la commande <code>installadm create-service</code> . Reportez-vous au <a href="#">Chapitre 2, "Configuration d'un serveur d'installation IA"</a> .

TABLEAU C-1 Comparaison des tâches JumpStart et du programme d'installation automatisée (Suite)

Tâche	JumpStart	Programme d'installation automatisée
Ajoutez des clients à l'installation.	Utilisez la commande <code>add_install_client(1M)</code> .	Utilisez la commande <code>installadm create-client</code> . Reportez-vous au <a href="#">Chapitre 7</a> , “Installation de systèmes clients”.
Spécifiez les instructions d'installation.	Utilisez les fichiers de profil.	Utilisez les fichiers manifest IA. Reportez-vous au <a href="#">Chapitre 4</a> , “Spécification des instructions d'installation”.
Spécifiez les personnalisations des clients.	Utilisez les fichiers de règles pour associer les clients avec les fichiers de profil.	Utilisez la commande <code>installadm set-criteria</code> pour associer les clients avec les fichiers manifest IA. Reportez-vous au <a href="#">Chapitre 3</a> , “Personnalisation des installations”.
Spécifiez la configuration du client post-installation.	Utilisez des scripts de fin et des fichiers <code>sysidcfg(4)</code> .	Utilisez des fichiers manifest SMF qui sont associés à des fichiers manifest IA spécifiques. Reportez-vous au <a href="#">Chapitre 5</a> , “Configuration du système client”.

## Conversion des règles JumpStart en critères IA

Les critères de client IA peuvent être spécifiés selon les deux méthodes suivantes. Les directives présentées dans le tableau suivant s'appliquent dans les deux cas. Reportez-vous au [Chapitre 3](#), “Personnalisation des installations”.

- En tant qu'options `-c` de la commande `installadm add-manifest` ou en tant qu'options `-c` ou `-a` de la commande `installadm set-criteria`
- Dans les fichiers de critères

## Comparaison entre les mots-clés de règles et les directives de critères

Le tableau ci-dessous compare les mots-clés de règles JumpStart avec les directives de critères IA. Le programme d'installation automatisée utilise ces critères pour appliquer les informations de fichiers manifest IA et SC (configuration système) correctes à un client particulier. Les mots-clés de règles JumpStart qui ne sont pas répertoriés dans ce tableau ne disposent pas d'une directive IA comparable, ou leur fonctionnalité ne s'applique pas au programme d'installation automatisée.

TABLEAU C-2 Comparaison des mots-clés de fichier de règles JumpStart et des directives de critères IA

Mot-clé de fichier de règles JumpStart	Directives de fichier de critères IA
any	Pour les systèmes client qui ne correspondent à aucun critère de sélection, le service d'installation IA fournit un fichier manifest IA par défaut.
arch sparc	Option de commande : -c cpu=sparc Fichier de critères : <ai_criteria name="cpu"> <value>sparc</value> </ai_criteria>
hostaddress 10.6.68.127	Option de commande : -c ipv4=10.6.68.127 Fichier de critères : <ai_criteria name="ipv4"> <value>10.6.68.127</value> </ai_criteria>
hostname solaris	Afin d'identifier de manière unique un hôte dans le programme d'installation automatisée, utilisez l'adresse IP comme indiqué ci-dessus, ou utilisez l'adresse MAC. Option de commande : -c mac=0:3:ba:33:9d:b6 Fichier de critères : <ai_criteria_name="mac"> <value>0:3:ba:33:9d:b6</value> </ai_criteria>
karch i86pc	Option de commande : -c arch=i86pc Fichier de critères : <ai_criteria name="arch"> <value>i86pc</value> </ai_criteria>
memsize 2048	Option de commande : -c mem=2048 Fichier de critères : <ai_criteria name="mem"> <value>2048</value> </ai_criteria>

**TABLEAU C-2** Comparaison des mots-clés de fichier de règles JumpStart et des directives de critères IA  
(Suite)

Mot-clé de fichier de règles JumpStart	Directives de fichier de critères IA
model SUNW, Sun-Fire-T200	Option de commande : -c platform=SUNW, Sun-Fire-T200  Fichier de critères :  <ai_criteria_name="platform"> <value>SUNW, Sun-Fire-T200</value> </ai_criteria>
network 10.0.0.1	Utilisez ipv4 avec une plage de valeurs.  Option de commande : -c ipv4=10.0.0.1-10.0.0.64  Fichier de critères :  <ai_criteria name="ipv4"> <range>10.0.0.1 10.0.0.64</range> </ai_criteria>

## Conversion des fichiers de règles en critères

Cette section contient des exemples de règles JumpStart converties en critères IA. Les sous-commandes `add-manifest` et `set-criteria` associent les critères client avec les fichiers manifest IA spécifiés. Reportez-vous à la rubrique [“Association d'instructions d'installation spécifiques au client à des services d'installation”](#) à la page 26.

### Exemple de fichier de règles JumpStart

Dans un fichier de règles JumpStart, la première colonne contient les mots-clés de règles et les valeurs de règles qui permettent d'identifier les systèmes client, la deuxième colonne correspond au script de début, la troisième colonne correspond au profil et la quatrième colonne contient le script de fin.

L'exemple de fichier de règles JumpStart suivant identifie les trois groupes de systèmes, et un profil différent est affecté à chaque groupe. Dans cet exemple, aucun script de début ou de fin n'est spécifié pour aucun des trois groupes client. Cet exemple de fichier de règles indique les règles suivantes :

- Tous les systèmes SPARC avec une taille de mémoire comprise entre 4 096 et 2 048 Mo seront installés à l'aide des spécifications du profil `lx_prof`.
- Tous les systèmes x86 seront installés à l'aide des spécifications du profil `prog_prof`.
- Tous les systèmes qui ne sont pas abordés dans les règles ci-dessus seront installés à l'aide des spécifications du profil `generic_prof`.

```

memsize 2048-4096 && model SUNW - lx_prof -
karch i86pc - prog_prof -
any - generic_prof -

```

## Spécifications des critères IA équivalents

Cette section présente les commandes IA qui sont comparables aux trois règles de l'exemple JumpStart précédent. Dans ces exemples, le profil `lx_prof` a été converti sous la forme du fichier manifest `manifest_lx.xml`, et le profil `prog_prof` a été converti sous la forme du fichier manifest `manifest_prog.xml`. Les clients qui ne correspondent à aucun des critères spécifiés utilisent le fichier manifest IA par défaut.

Dans l'exemple suivant, tous les systèmes SPARC avec une taille de mémoire comprise entre 2 048 Mo et 4 096 Mo seront installés à l'aide des spécifications du fichier manifest IA `manifest_lx.xml`.

```

# installadm add-manifest -m ./manifest_lx.xml -n s11-sparc \
-c arch="sparc" -c mem="2048-4096"

```

Vous pouvez également placer les critères dans un fichier et référencer le fichier. La commande suivante est l'équivalent de la commande précédente si le contenu du fichier `criteria_lx.xml` est comme indiqué.

```

# installadm add-manifest -m ./manifest_lx.xml -n s11-sparc \
-C ./criteria_lx.xml

```

```

<ai_criteria_manifest>
  <ai_criteria name="cpu">
    <value>sparc</value>
  </ai_criteria>
  <ai_criteria name="mem">
    <range>
      2048
      4096
    </range>
  </ai_criteria>
</ai_criteria_manifest>

```

Dans l'exemple suivant, tous les systèmes x86 seront installés à l'aide des spécifications du fichier manifest IA `manifest_prog.xml`.

```

# installadm set-criteria -m ./manifest_prog.xml -n s11-x86 \
-c arch="i86pc"

```

## Conversion d'un profil JumpStart en fichier manifest IA

Le tableau suivant compare les mots-clés du profil JumpStart avec les directives du fichier manifest IA. Le programme d'installation automatisée utilise les fichiers manifest XML pour définir l'installation client. Reportez-vous au [Chapitre 4, “Spécification des instructions d'installation”](#). Les mots-clés du profil JumpStart qui ne sont pas répertoriés ne disposent pas d'une directive IA comparable, ou leur fonctionnalité ne s'applique pas au programme d'installation automatisée.

TABLEAU C-3 Comparaison des mots-clés du fichier du profil JumpStart et des directives du fichier manifest IA

Mot-clé du fichier de règles JumpStart	Directives du fichier manifest IA
boot_device c0t0d0s0 update	<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;disk_name name="c0t0d0" name_type="ctd"/&gt;       &lt;slice name="0" is_root="true" force="true"/&gt;     &lt;/disk&gt;   &lt;/target_device&gt; &lt;/target&gt;</pre> <p>La deuxième valeur de jeton (update pour les systèmes SPARC et preserve pour les systèmes x86) pour ce mot-clé n'est pas prise en charge dans le programme d'installation automatisée. Dans le programme d'installation automatisée, la mémoire EEPROM des systèmes SPARC est toujours mise à jour pour le périphérique cible spécifié, de sorte que le système installé s'initialise automatiquement à partir de ce périphérique. Sur les systèmes x86, le microprogramme n'est jamais mis à jour.</p>
bootenv	Un environnement d'initialisation est créé automatiquement lors de l'installation du SE Oracle Solaris.
cluster SUNWCXall	<p>Oracle Solaris 11 ExpressOS utilise les packages de groupe pkg(5). Les packages de groupe sont spécifiés à l'instar de tout autre package spécifié dans le fichier manifest. Le fichier manifest IA par défaut inclut les packages nécessaires pour une installation Oracle Solaris 11 Express standard. Vous pouvez personnaliser cette liste de packages.</p> <pre>&lt;software_data action="install" type="IPS"&gt;   &lt;name&gt;pkg:/entire&lt;/name&gt;   &lt;name&gt;pkg:/babel_install&lt;/name&gt; &lt;/software_data&gt;</pre>



**TABLEAU C-3** Comparaison des mots-clés du fichier du profil JumpStart et des directives du fichier manifest IA *(Suite)*

Mot-clé du fichier de règles JumpStart	Directives du fichier manifest IA
fdisk c0t3d0 solaris maxfree	<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;disk_name name="c0t3d0" name_type="ctd"/&gt;       &lt;partition name="1" part_type="SOLARIS"&gt;     &lt;/disk&gt;   &lt;/target_device&gt; &lt;/target&gt;</pre> <p>Reportez-vous à la section <a href="#">“Configuration du partitionnement sur un client x86”</a> à la page 51.</p>
filesystem	<p>Dans JumpStart, ce mot-clé <code>filesystem</code> crée la tranche VTOC et le système de fichiers UFS sur cette tranche. Le programme d'installation automatisée crée les systèmes de fichiers ZFS, et non pas les systèmes de fichiers UFS. Pour créer des tranches VTOC dans le programme d'installation automatisée, utilisez l'élément <code>&lt;slice&gt;</code>. Reportez-vous à la rubrique <a href="#">“Configuration de tranches sur un disque”</a> à la page 55.</p>
geo	<p>Les régions géographiques pour la prise en charge des langues sont spécifiées par le biais des packages <code>pkg(5)</code>. Les packages de groupe sont spécifiés à l'instar de tout autre package spécifié dans le fichier manifest. Reportez-vous au mot-clé <code>cluster</code> JumpStart ci-dessus pour de plus amples informations.</p>
locale	<p>La prise en charge des paramètres régionaux est spécifiée par le biais des packages de groupe <code>pkg(5)</code>. Les packages de groupe sont spécifiés à l'instar de tout autre package spécifié dans le fichier manifest. Reportez-vous au mot-clé <code>cluster</code> ci-dessus pour de plus amples d'informations.</p>
package SUNWmysql51 add nfs golden:/packages/Solaris_10	<pre>&lt;software name="IPS"&gt;   &lt;software_data action="install" type="IPS"&gt;     &lt;name&gt;pkg:/entire&lt;/name&gt;     &lt;name&gt;pkg:/babel_install&lt;/name&gt;     &lt;name&gt;pkg:/database/mysql-51&lt;/name&gt;   &lt;/software_data&gt; &lt;/software&gt;</pre>
package SUNWmysql delete	<pre>&lt;software_data action="uninstall" type="IPS"&gt;   &lt;name&gt;pkg:/database/mysql-4&lt;/name&gt; &lt;/software_data&gt;</pre>
partitioning default partitioning existing partitioning explicit	<p>Si les directives <code>target_device</code> sont omises dans le fichier manifest IA, un partitionnement par défaut est utilisé. Reportez-vous à la rubrique <a href="#">“Configuration du partitionnement sur un client x86”</a> à la page 51.</p>

**TABLEAU C-3** Comparaison des mots-clés du fichier du profil JumpStart et des directives du fichier manifest IA  
(Suite)

Mot-clé du fichier de règles JumpStart	Directives du fichier manifest IA
usedisk	<pre>&lt;target&gt;   &lt;target_device&gt;     &lt;disk&gt;       &lt;disk_keyword key="boot_disk"/&gt;     &lt;/disk&gt;   &lt;/target_device&gt; &lt;/target&gt;</pre>

## Conversion des fichiers de profil en fichiers manifest IA

Cette section contient des exemples de spécifications de caractéristiques de profil JumpStart converties en directives de fichier manifest IA. Reportez-vous au [Chapitre 4, “Spécification des instructions d’installation”](#) pour plus d’informations sur les directives de fichier manifest IA.

### Exemple de fichier de profil JumpStart

Dans un profil JumpStart, la première colonne contient les mots-clés de profil, et la deuxième colonne contient les valeurs de profil.

L'exemple de profil JumpStart suivant indique ce qui suit :

- Les systèmes recevront une installation initiale.
- Les systèmes seront configurés en tant que systèmes autonomes.
- Sur chaque système, une partition `fdisk` est créée dans le plus grand espace disponible contigu du disque `c0t0d0`.
- Le groupe de logiciels Entire Distribution, `SUNWCall`, est installé sur le système.
- La suite StarOffice d’outils est installée sur le système.
- Les utilitaires de courrier électronique et de calendrier Evolution sont installés sur le système.

```
install_type initial_install
system_type standalone
fdisk c0t0d0 solaris maxfree
cluster SUNWCall
cluster SUNWCstaroffice
cluster SUNWCevo
```

### Directives de fichier manifest IA équivalentes

Les extraits de fichiers manifest IA suivants montrent comment vous pouvez convertir le profil JumpStart précédent en directives dans les fichiers IA.

L'exemple suivant indique au programme d'installation d'installer la version Oracle Solaris 11 Express sur le disque `c0t0d0` et de créer une partition SOLARIS sur la plus grande partie d'espace libre contigu. Reportez-vous à la section [“Configuration du partitionnement sur un client x86” à la page 51](#). L'exemple suivant spécifie une source de packages à installer et installe les deux packages de groupe obligatoires. Reportez-vous à la section [“Spécification des packages à installer” à la page 61](#) pour savoir pourquoi ces mêmes packages de groupe doivent être désinstallés. Enfin, cet exemple installe OpenOffice et Evolution.

```
<?xml version="1.0"?>
<!DOCTYPE auto_install SYSTEM "file:///usr/share/auto_install/ai.dtd">
<auto_install>
  <ai_instance name="sample_ai_manifest" auto_reboot="true">
    <target>
      <target_device>
        <disk>
          <disk_name name="c0t0d0" name_type="ctd"/>
          <partition name="1" part_type="SOLARIS">
        </disk>
      </target_device>
    </target>
    <software name="ips">
      <source>
        <publisher name="solaris">
          <origin name="http://pkg.oracle.com/solaris/release"/>
        </publisher>
      </source>
      <software_data type="IPS">
        <name>pkg:/entire</name>
        <name>pkg:/babel_install</name>
      </software_data>
      <software_data action="uninstall" type="IPS">
        <name>pkg:/babel_install</name>
        <name>pkg:/slim_install</name>
      </software_data>
      <software_data type="IPS">
        <name>pkg:/office/openoffice</name>
        <name>pkg:/mail/evolution</name>
      </software_data>
    </software>
  </ai_instance>
</auto_install>
```

