

Création d'une image d'installation Oracle® Solaris 11 personnalisée

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.

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. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de The Open Group.

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.

Table des matières

- Préface5**

- 1 Introduction à la création d'une image d'installation personnalisée9**
 - À propos du constructeur de distribution9
 - Types d'images Oracle Solaris 10
 - Processus de création d'images 11
 - Différences entre les archives SPARC et x86 12

- 2 Création d'une image d'installation personnalisée 13**
 - Configuration système requise pour la création d'images 13
 - Personnalisation des images 14
 - Échantillons de fichiers manifestes 14
 - ▼ Création et génération d'une image personnalisée 15
 - Modification du contenu du fichier manifeste 16
 - Création et utilisation de scripts personnalisés 23

- 3 Création d'une image25**
 - Commande `distro_const` 25
 - ▼ Création d'une image en une seule étape 26
 - ▼ Création d'une image en plusieurs étapes 27

Préface

Création d'une image d'installation Oracle Solaris 11 personnalisée fournit des instructions relatives à l'utilisation du constructeur de distribution Oracle Solaris pour créer des images d'installation Oracle Solaris personnalisées.

Informations connexes

La page *Installation des systèmes Oracle Solaris 11* fournit des instructions relatives à l'installation et à la configuration du système d'exploitation (SE) Oracle Solaris à l'aide de l'une des méthodes suivantes :

- Une image LiveCD
- Un programme d'installation en mode texte interactif
- La fonction de programme d'installation automatisée Oracle Solaris
- L'outil de configuration système interactif SCI Tool Oracle Solaris
- L'outil de configuration système de ligne de commande sysconfig(1M)

La page *Création et administration d'environnements d'initialisation Oracle Solaris 11* décrit la gestion de plusieurs environnements d'initialisation sur votre système Oracle Solaris, y compris les zones non globales.

Le Chapitre 6, “Gestion des services (présentation)” du manuel *Administration d'Oracle Solaris : Tâches courantes*, décrit la fonction SMF (Service Management Facility, Utilitaire de gestion des services) d'Oracle Solaris. Vous pouvez utiliser les profils SMF pour configurer votre système.

La page de manuel [pkg\(5\)](#) décrit la fonction IPS (Image Packaging System) d'Oracle Solaris, qui vous permet de stocker et d'extraire les packages logiciels pour l'installation. La page de manuel [pkg\(1\)](#) explique comment installer les packages IPS.

Reportez-vous à la documentation relative à l'administration système d'Oracle Solaris 11 pour plus d'informations sur l'administration de systèmes Oracle Solaris 11.

La page *Transition de JumpStart d'Oracle Solaris 10 au programme d'installation automatisée d'Oracle Solaris 11* fournit des informations destinées à vous aider à migrer de JumpStart au programme d'installation automatisée, qui sont tous deux des fonctions d'installation automatisée d'Oracle Solaris.

Accès au support technique Oracle

Les clients Oracle ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> ou le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> adapté aux utilisateurs malentendants.

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.
AaBbCc123	Ce que vous entrez, par opposition à ce qui s'affiche à l'écran.	<code>nom_machine%</code> su 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> .
<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. Remarque : 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#

Introduction à la création d'une image d'installation personnalisée

Les administrateurs système et les développeurs d'applications peuvent utiliser le constructeur de distribution pour créer des images d'installation Oracle Solaris personnalisées.

- Si vous n'avez pas créé d'images d'installation personnalisée auparavant, reportez-vous à la section [“À propos du constructeur de distribution” à la page 9.](#)
- Si vous êtes prêt à créer des images personnalisées, rendez-vous à la section [“Configuration système requise pour la création d'images” à la page 13.](#)

À propos du constructeur de distribution

Le constructeur de distribution est un outil de ligne de commande pour la création d'images Oracle Solaris préconfigurées. Il récupère le fichier manifeste XML en entrée et génère une image en fonction des paramètres spécifiés dans le fichier manifeste.

Le constructeur de distribution peut générer une image ISO, qui est un fichier archive, également appelé image disque, d'un disque optique dans un format défini par l'Organisation internationale de normalisation (ISO). Vous pouvez également créer une image USB basé sur une image ISO générée. Cependant, contrairement aux images ISO, une image USB peut uniquement être créée et utilisée sur les systèmes x86.

Prenez note des remarques suivantes :

- Selon la configuration d'image, les images ISO ou USB peuvent être initialisables.
- Les images ISO et USB peuvent être installées sur un système ou exécutées dans un environnement multimédia en direct.
- Une image ISO peut être gravée sur un CD ou un DVD.
- Une image USB peut être copiée vers un lecteur Flash.
- Les images ISO et USB peuvent être publiées sur Internet.

Le constructeur de distribution crée une image USB qui peut fonctionner avec plusieurs types de périphériques de mémoire flash, dans la mesure où la version Oracle Solaris prend en charge le pilote de périphérique. L'utilitaire `usbcopy` doit être utilisé pour copier une image USB vers un lecteur flash USB. Cet utilitaire est disponible dans le package `distribution-creator`.

Types d'images Oracle Solaris

Vous pouvez vous servir du constructeur de distribution pour créer les types d'images Oracle Solaris suivants.

- **Image LiveCD x86 Oracle Solaris** : vous pouvez créer une image ISO x86 comparable à l'image LiveCD distribuée comme version Oracle Solaris. En outre, vous pouvez modifier le contenu de cette image ISO en ajoutant ou supprimant des packages. Vous pouvez également modifier les paramètres par défaut pour l'environnement initialisé de manière à créer une image ISO ou USB personnalisée.

Remarque – En fonction de la taille des packages inclus dans l'image LiveCD, l'image LiveCD peut être une image LiveDVD.

Pour plus d'informations sur les installations LiveCD, reportez-vous au [Chapitre 3](#), “Utilisation de LiveCD” du manuel *Installation des systèmes Oracle Solaris 11*.

- **Image d'installation en mode texte Oracle Solaris x86 ou SPARC** : vous pouvez créer une image ISO SPARC ou x86 pouvant être utilisée pour effectuer une installation en mode texte du système d'exploitation Oracle Solaris. Le programme d'installation en mode texte peut être utilisé sur des systèmes ne nécessitant pas de carte graphique.

Remarque – Une installation en mode texte n'installe *pas* tous les packages logiciels inclus lors de l'installation à partir du LiveCD. Par exemple, le programme d'installation en mode texte n'installe pas de bureau. Après une installation en mode texte, vous pouvez ajouter des packages supplémentaires, tels que le package `solaris-desktop`.

Pour plus d'informations sur les installations en mode texte, reportez-vous au [Chapitre 4](#), “Utilisation du programme d'installation en mode texte” du manuel *Installation des systèmes Oracle Solaris 11*.

- **Image ISO x86 ou SPARC pour l'installation automatisée** : la version Oracle Solaris inclut l'outil d'installation automatisée. Le programme d'installation automatisée (IA) permet d'automatiser l'installation du SE Oracle Solaris sur un ou plusieurs systèmes SPARC et x86 sur un réseau. Les installations peuvent différer en termes d'architecture, de packages installés, de capacité du disque et d'autres paramètres. Le programme d'installation automatisée utilise une image ISO d'installation automatisée SPARC ou x86 pour installer le SE Oracle Solaris sur les systèmes clients. Vous pouvez vous servir du constructeur de

distribution pour créer une image ISO d'installation automatisée SPARC permettant d'installer le SE Oracle Solaris sur des clients SPARC, ou pour créer une image ISO d'installation automatisée x86 permettant d'installer le SE Oracle Solaris sur des clients x86.

Pour plus d'informations sur l'utilisation du programme d'installation automatisée, reportez-vous à la [Partie III, “Installation à l’aide d’un serveur d’installation” du manuel *Installation des systèmes Oracle Solaris 11*](#).

Processus de création d'images

Le constructeur de distribution crée des images en fonction des paramètres indiqués dans des fichiers XML, appelés *fichiers manifestes*. Les fichiers manifestes contiennent les spécifications relatives au contenu et aux paramètres des images ISO créées à l'aide du constructeur de distribution. Le constructeur de distribution contient des exemples de fichiers manifestes qui peuvent être utilisés pour créer une image ISO d'installation automatisée LiveCD, x86 ou SPARC personnalisée, ou une image d'installation en mode texte x86 ou SPARC.

Tous les champs des fichiers manifestes fournissent des valeurs par défaut, prédéfinies, qui créent le type d'image dont vous avez besoin. Vous pouvez modifier les champs dans le fichier manifeste afin de personnaliser davantage l'image résultante. Par exemple, vous pouvez modifier l'élément cible dans le manifeste pour indiquer un emplacement différent pour la zone de compilation où l'image peut être créée. En outre, vous pouvez vérifier l'éditeur spécifié et vous assurer que le système utilisé peut contacter cet éditeur pour télécharger les packages nécessaires à création de l'image. Si nécessaire, vous pouvez modifier l'élément de nom du logiciel pour spécifier un éditeur et un emplacement de référentiel différents. Pour plus d'instructions, reportez-vous à la section [“Personnalisation des images” à la page 14](#).

Vous pouvez également créer des *scripts personnalisés* pour modifier votre image d'installation. Vous pouvez ensuite ajouter des points de contrôle au fichier manifeste pour exécuter ces scripts personnalisés. Pour plus d'informations, reportez-vous à la section [“Création et utilisation de scripts personnalisés” à la page 23](#).

Le package du constructeur de distribution comprend également un utilitaire de ligne de commande, la commande `distro_const`, qui interprète les spécifications du fichier manifeste et génère l'image. Une fois l'empreinte de l'image modifiée selon vos besoins dans un fichier manifeste, exécutez la commande `distro_const` pour construire votre image. Pour plus d'informations, reportez-vous au [Chapitre 3, “Création d'une image”](#).

Vous pouvez utiliser les options offertes par la commande `distro_const` pour arrêter et redémarrer le processus de création à différentes étapes du processus de génération d'image, afin de contrôler et de déboguer l'image en cours de création. Ce processus d'arrêt et de redémarrage au cours de la création est appelé *création de points de contrôle*. La création de points de contrôle est facultative. Les valeurs par défaut des points de contrôle sont indiquées dans chaque fichier manifeste. Pour obtenir davantage d'instructions, reportez-vous à la section [“Création d'une image en plusieurs étapes” à la page 27](#) ou à la page de manuel `distro_const(1M)`.

Différences entre les archives SPARC et x86

L'archive root pour les images x86 diffère de l'archive root pour les images SPARC. L'archive whole root, ou `boot_archive`, pour les images x86, est un système de fichiers UFS compressé à l'aide de lzma. La plate-forme SPARC ne prend pas en charge la compression de l'archive whole root de cette manière. Au lieu de cela, les archives SPARC utilisent DCFS, qui compresse chaque fichier individuellement. Ces fichiers compressés séparément peuvent nécessiter un traitement spécifique dans le fichier manifeste. Pour plus d'instructions, reportez-vous au champ `<boot_archive_contents>` de la page de manuel `dc_manifest(4)`.

Création d'une image d'installation personnalisée

Vérifiez la configuration système requise et créez une image d'installation personnalisée en suivant les procédures décrites dans ce chapitre.

Configuration système requise pour la création d'images

Pour utiliser le constructeur de distribution, votre système doit répondre à la configuration requise suivante.

TABLEAU 2-1 Configuration système requise

Configuration requise	Description
Espace disque	La taille minimale recommandée pour l'espace de travail du constructeur de distribution est de 8 Go. Vérifiez que vous disposez de suffisamment d'espace sur votre système pour utiliser le constructeur de distribution.
Version d'Oracle Solaris	<p>Un système d'exploitation (SE) Oracle Solaris doit être installé sur votre système. Veuillez prendre en compte les points importants suivants :</p> <ul style="list-style-type: none"> ■ Votre système installé doit avoir accès au réseau. Le constructeur de distribution accède aux référentiels IPS (Image Packaging System, système d'emballage d'image) disponibles sur le réseau pour récupérer les packages pour l'image ISO. Vous devez disposer d'un accès réseau aux référentiels spécifiés dans le fichier manifeste. ■ Lorsque vous utilisez le constructeur de distribution, vous pouvez uniquement créer des images SPARC sur un système SPARC et des images x86 sur un système x86. ■ La version Oracle Solaris sur votre système doit être la même que celle des images que vous utilisez avec le constructeur de distribution. <p>Remarque – Vous devez vous connecter avec le rôle root pour exécuter le constructeur de distribution.</p>

TABLEAU 2-1 Configuration système requise (Suite)

Configuration requise	Description
Packages requis	Package <code>distribution-creator</code> , qui contient l'application du constructeur de distribution.

Personnalisation des images

Le constructeur de distribution crée des images en fonction des paramètres indiqués dans des fichiers XML, appelés *fichiers manifestes*. Les fichiers manifestes contiennent les spécifications relatives au contenu et aux paramètres des images ISO créées à l'aide du constructeur de distribution. Le package `distribution-creator` fournit des exemples de fichiers manifestes qui peuvent être utilisés pour créer une image ISO d'installation automatisée LiveCD, x86 ou SPARC, ou une image d'installation en mode texte x86 ou SPARC.

Les éléments présents dans chaque fichier manifeste fournissent des valeurs par défaut, prédéfinies, qui créent le type d'image ISO dont vous avez besoin. Vous pouvez modifier manuellement ces éléments prédéfinis dans un fichier manifeste afin de personnaliser davantage l'image obtenue. En outre, vous pouvez créer des scripts personnalisés pour encore plus modifier votre image. Ensuite, faites référence aux nouveaux scripts dans le fichier manifeste.

Échantillons de fichiers manifestes

Le package `distribution-creator` fournit les échantillons de fichiers manifestes suivants.

TABLEAU 2-2 Échantillons de fichiers manifestes

Type de fichier manifeste	Emplacement du fichier manifeste	Description
Image ISO LiveCD x86	<code>/usr/share/distro_const/dc_livecd.xml</code>	Crée une image ISO comparable au LiveCD Oracle Solaris
Image d'installation en mode texte x86	<code>/usr/share/distro_const/dc_text_x86.xml</code>	Crée une image ISO pouvant être utilisée pour effectuer une installation en mode texte du système d'exploitation Oracle Solaris x86
Image d'installation en mode texte SPARC	<code>/usr/share/distro_const/dc_text_sparc.xml</code>	Crée une image ISO pouvant être utilisée pour effectuer une installation en mode texte du système d'exploitation Oracle Solaris SPARC
Image ISO d'installation automatisée x86	<code>/usr/share/distro_const/dc_ai_x86.xml</code>	Crée une image ISO d'installation automatisée x86 pour l'installation automatisée du SE Oracle Solaris sur des clients x86

TABLEAU 2-2 Échantillons de fichiers manifestes (Suite)

Type de fichier manifeste	Emplacement du fichier manifeste	Description
Image ISO d'installation automatisée SPARC	/usr/share/distro_const/ dc_ai_sparc.xml	Crée une image ISO d'installation automatisée SPARC pour l'installation automatisée du SE Oracle Solaris sur des clients SPARC

▼ Création et génération d'une image personnalisée

- 1 **Téléchargez le package `distribution-creator`, qui contient l'application du constructeur de distribution et des échantillons de fichiers manifestes.**

Vous pouvez utiliser le gestionnaire de packages pour installer le package requis. Le gestionnaire de packages est disponible dans la barre de menu sur le bureau du système d'exploitation Oracle Solaris. Dans la barre de menu, cliquez sur Système>Administration>Gestionnaire de packages.

Vous pouvez également utiliser les commandes IPS (la suivante, par exemple) pour installer ce package :

```
# pkg install distribution-creator
```

- 2 **Copiez un des exemples de fichiers manifestes et créez un fichier manifeste personnalisé avec un nouveau nom de fichier.**

Vous ferez référence au fichier manifeste par son nom lorsque vous utiliserez la commande `distro_const` pour créer une image.

Remarque – Sauvegardez toujours le fichier manifeste d'origine et les scripts par défaut avant de les copier.

- 3 **Modifiez les éléments du fichier manifeste selon vos besoins.**

Par exemple, vous pouvez modifier l'élément `target` dans le manifeste pour indiquer un emplacement différent pour la zone de compilation où l'image peut être créée. Ainsi, vous pouvez vérifier l'éditeur afin de vous assurer que votre système peut contacter cet éditeur pour télécharger les packages nécessaires à la création de l'image. Si nécessaire, vous pouvez modifier l'élément de nom du logiciel pour spécifier un éditeur et un emplacement de référentiel différents.

Pour plus d'informations, reportez-vous à la section [“Modification du contenu du fichier manifeste” à la page 16](#) et à la page de manuel `dc_manifest(4)`.

- 4 **(Facultatif) Créez des scripts personnalisés pour modifier davantage l'image.**

Si vous créez de nouveaux scripts, mettez à jour les références de script dans la section d'exécution du fichier manifeste.

Pour obtenir davantage d'instructions, reportez-vous à la section [“Création et utilisation de scripts personnalisés”](#) à la page 23.

5 Exécutez l'utilitaire `distro_const` pour créer une image.

Pour obtenir davantage d'instructions, reportez-vous au [Chapitre 3, “Création d'une image”](#).

Modification du contenu du fichier manifeste

Tous les champs des fichiers manifestes contiennent des valeurs par défaut, prédéfinies, qui créent le type d'image ISO dont vous avez besoin. Vous pouvez modifier manuellement ces champs prédéfinis dans un fichier manifeste afin de personnaliser davantage l'image obtenue.

Selon l'échantillon de fichier manifeste que vous sélectionnez, les principaux éléments sont les suivants.

TABLEAU 2-3 Éléments du fichier manifeste

Élément	Description
<code><distro name="Oracle_Solaris_Text_X86" add_timestamp="false"></code>	Spécifie le nom de l'image avec un horodatage facultatif
<code><boot_mods></code>	Spécifie les modifications du menu GRUB pour l'image
<code><target></code>	Définit le jeu de données ZFS dans lequel l'image est créée
<code><software name="transfer-ips-install" type="IPS"></code>	Spécifie la source des packages logiciels à installer
<code><software_data action="install"></code>	Répertorie les packages à installer
<code><software_data action="uninstall"></code>	Répertorie les packages à désinstaller
<code><software name="set-ips-attributes"></code>	Définit des attributs différents pour l'IPS une fois l'installation terminée
<code><software name="ba-init"></code>	Spécifie le contenu de l'archive d'initialisation Attention – Modifiez cet élément avec précaution. Si l'archive d'initialisation est incorrecte, le système installé ne démarrera pas.
<code><execution stop_on_error="true"></code> <code><checkpoint name="transfer-ips-install"/></code>	Répertorie les points de contrôle de compilation
<code><configuration name="pre-pkg-img-mod" type="sysconf" source="/etc/svc/profile/generic_limited_net.xml"></code>	Spécifie les services SMF à appliquer à des médias lors de la création Attention – Modifiez cet élément le plus rarement possible.

Indication d'un titre pour l'image

Utilisez l'élément suivant pour fournir un nom personnalisé ou par défaut à l'image que vous allez créer.

```
<distro name="Oracle_Solaris_Text_X86" add_timestamp="false">
```

Si vous avez l'intention de créer une série de versions d'une image et de conserver les images incrémentielles, vous pouvez définir la variable d'horodatage sur "true" : un horodatage sera automatiquement ajouté au nom de chaque image.

Si vous avez besoin de spécifier un proxy HTTP, supprimez les marques de commentaire de l'élément `distro name` incluant la variable de proxy, puis saisissez l'emplacement du proxy.

Modification du menu d'initialisation

L'élément de menu d'initialisation indique les modifications du menu d'initialisation à appliquer à l'image.

Dans l'exemple ci-dessous, un menu d'initialisation spécialisé avec le titre "boot1" sera appliqué à l'image. L'attribut de délai d'attente spécifie la durée avant l'activation automatique de l'entrée d'initialisation par défaut.

```
<boot_mods title="boot1" timeout="5">
```

Dans l'élément de menu d'initialisation, vous pouvez ajouter des entrées individuelles du menu d'initialisation en ajoutant un nouvel élément `boot_entry` pour chaque nouvelle entrée. Les entrées sont ajoutées séquentiellement au menu d'initialisation, dans un ordre défini par la valeur d'attribut `insert_at` "start" ou "end" pour chaque entrée d'initialisation.

Remarque – Ajoutez les nouvelles entrées avant l'entrée "with magnifier" existante.

Reportez-vous à l'exemple ci-après d'un élément `boot_entry` individuel.

```
<boot_entry>
  <title_suffix>with screen reader</title_suffix>
  <kernel_args>-B assistive_tech=reader</kernel_args>
</boot_entry>
```

Pour plus d'informations, reportez-vous à la page de manuel `dc_manifest(4)`.

Spécification de la zone de compilation

Vous pouvez personnaliser l'élément `target`. Cet élément définit le jeu de données ZFS à utiliser pour la création. Ce jeu de données correspond à la zone où l'image sera créée. Vous devez saisir un emplacement de jeu de données valide. Vérifiez la zone de compilation par défaut afin de vous assurer que la compilation ne détruira pas le contenu que vous devez conserver sur votre système. Modifiez la zone de compilation si nécessaire.

Remarque – Le nom du système de fichiers ne doit pas inclure le nom du zpool.

Voyez l'exemple suivant :

```
<target>
  <logical>
    <zpool action="use_existing" name="rpool">
      <dataset>
        <filesystem name="dc/sample-dataset-location"
          action="preserve"/>
      </dataset>
    </zpool>
  </logical>
</target>
```

Spécification de l'éditeur

L'élément suivant spécifie un éditeur auprès duquel le constructeur de distribution peut obtenir des packages à télécharger et à utiliser pour créer l'image.

```
<software name="transfer-ips-install">
```

Dans l'élément source de cette section, modifiez le nom de l'éditeur et le nom d'origine afin de spécifier l'éditeur à utiliser et l'emplacement du référentiel de package. Plusieurs éditeurs peuvent être répertoriés. Quand le constructeur de distribution tente de localiser les packages à installer, les éditeurs sont parcourus dans l'ordre dans lequel ils sont répertoriés ici.

Si les copies miroir d'un éditeur doivent être spécifiées, supprimez les commentaires et modifiez l'élément de nom de miroir.

Voyez l'exemple suivant :

```
<source>
  <publisher name="publisher1">
    <origin name="http://example.oracle.com/primary-pub"/>
    <mirror name="mirror.example.com"/>
  </publisher>
  <publisher name="publisher2">
    <origin name="http://example2.com/dev/solaris"></origin>
  </publisher>
  <publisher name="publisher3.org">
    <origin name="http://example3.com/dev"></origin>
  </publisher>
</source>
```

Pour plus d'informations sur l'utilisation des éditeurs, reportez-vous à la section [Ajout et mise à jour de packages logiciels Oracle Solaris 11](#).

Spécification des packages à installer

L'élément `software_data` avec l'attribut `install` répertorie l'ensemble des packages à installer afin de créer un type d'image particulier, en fonction du manifeste utilisé. Par exemple, le manifeste `dc_livecd.xml` répertorie les packages requis pour créer une image LiveCD. Chaque balise de nom indique un nom de package ou le nom d'un package de groupe contenant plusieurs packages.

```
<software_data action="install">
  <name>pkg:/group/system/solaris-desktop</name>
  <name>pkg:/system/install/gui-install</name>
  <name>pkg:/system/install/media/internal</name>
</software_data>
```

Si vous disposez de packages que vous souhaitez ajouter à l'image, ajoutez le nom des packages en ajoutant une balise de nom pour chaque package.

Par défaut, la version la plus actuelle du package disponible dans le référentiel spécifié est installée. Si une autre version est requise, ajoutez le numéro de version à la référence du package en utilisant le format suivant :

```
<name>pkg:/group/system/solaris-desktop@0.5.11-0.build#</name>
```

Remarque – Les packages avec une version particulière spécifiée peuvent ne pas être installés si d'autres packages avec une version en conflit est installée. Pour plus d'informations, reportez-vous à la page de manuel `pkg(5)`

EXEMPLE 2-1 Ajout de packages et d'éditeurs supplémentaires

Dans cet exemple, un deuxième éditeur, `mypublisher`, est spécifié. D'autres packages, `mypackage1` et `mypackage2`, sont également spécifiés.

Au cours du processus de compilation, les éditeurs sont vérifiés dans l'ordre dans lequel ils sont répertoriés. Si les packages ne sont pas disponibles auprès du premier éditeur, l'éditeur suivant est parcouru à la recherche des packages spécifiés.

```
<software name="transfer-ips-install" type="IPS">
  <destination>
    <xi:include xmlns:xi="http://www.w3.org/2003/XInclude"
      href="/usr/share/distro_const/lang_facets.xml"/>
  </destination>
  <source>
    <publisher name="solaris">
      <origin name="http://pkg.oracle.com/solaris/release"/>
    </publisher>
    <publisher name="mypublisher">
      <origin name="http://mypublisher.company.com"/>
    </publisher>
  </source>
</software_data action="install">
```

EXEMPLE 2-1 Ajout de packages et d'éditeurs supplémentaires (Suite)

```

<name>pkg:/group/system/solaris-large-server</name>
<name>pkg:/system/install/text-install</name>
<name>pkg:/system/install/media/internal</name>
<name>pkg:/mypackage1</name>
<name>pkg:/mypackage2</name>
</software_data>
</software>

```

Spécification des packages à désinstaller

L'élément `software_data` avec l'attribut `uninstall` peut être utilisé pour désinstaller un package ou une définition de package de groupe.

Dans l'exemple suivant, `solaris-desktop` est le nom d'un package de groupe qui contient plusieurs packages individuels.

```

<software_data action="uninstall">
  <name>pkg:/group/system/solaris-desktop</name>
</software_data>

```

Vous pouvez désinstaller un package de groupe. La désinstallation d'un package de groupe signifie que seule la définition de groupe est effectivement désinstallée. Les packages individuels précédemment installés dans le cadre de ce groupe ne sont pas désinstallés. Cependant, vous pouvez désinstaller les packages individuels sans désinstaller le package de groupe. Conserver le package de groupe peut être utile pour référence. Vous pouvez également utiliser la balise de nom pour désinstaller un package. Ajoutez les packages supplémentaires à désinstaller à la fin de la section de désinstallation.

Spécification d'un éditeur pour un système installé

L'élément suivant s'applique à un système une fois qu'il a été installé avec l'image créée à l'aide du constructeur de distribution.

```

<software name="set-ips-attributes">

```

Fournissez le nom de l'éditeur et des balises de nom de copie miroir facultatives pour spécifier l'endroit où le système installé peut accéder à d'autres packages à télécharger et à installer.

Vous pouvez également définir des attributs IPS dans cet élément. Pour plus d'informations sur les propriétés IPS, reportez-vous à la page de manuel `pkg(1)`.

Définition de points de contrôle de compilation

L'élément d'exécution dans le fichier manifeste répertorie une série de points de contrôle exécutés au cours du processus de compilation d'une image. Les points de contrôle sont exécutés dans l'ordre dans lequel ils sont répertoriés dans cette section. Les points de contrôle par défaut nécessaires à la compilation de l'image d'installation par défaut sont inclus dans chaque fichier manifeste.

Toutes les balises de nom de point de contrôle incluent l'attribut `mod-path`, qui spécifie l'emplacement du script du point de contrôle.

Certaines balises de points de contrôle par défaut incluent des arguments avec des valeurs par défaut fournies. L'exemple de point de contrôle suivant extrait de l'échantillon de fichier manifeste `dc_ai_sparc.xml` crée l'archive d'initialisation pour la compilation de l'image et renvoie à un script qui accomplira cette tâche. L'exemple de point de contrôle inclut également des champs d'argument avec des valeurs spécifiques fournies pour chaque argument.

```
<checkpoint name="ba-arch"
  desc="Boot Archive Archival"
  mod_path="solaris_install/distro_const/checkpoints/
  boot_archive_archive"
  checkpoint_class="BootArchiveArchive">
  <kwargs>
    <arg name="size_pad">0</arg>
    <arg name="bytes_per_inode">0</arg>
    <arglist name="uncompressed_files">
      <argitem>etc/svc/repository.db</argitem>
      <argitem>etc/name_to_major</argitem>
      <argitem>etc/minor_perm</argitem>
      <argitem>etc/driver_aliases</argitem>
      <argitem>etc/driver_classes</argitem>
      <argitem>etc/path_to_inst</argitem>
      <argitem>etc/default/init</argitem>
      <argitem>etc/nsswitch.conf</argitem>
      <argitem>etc/passwd</argitem>
      <argitem>etc/shadow</argitem>
      <argitem>etc/inet/hosts</argitem>
    </arglist>
  </kwargs>
</checkpoint>
```

Comme indiqué dans cet exemple, l'élément `<kwargs>` contient des arguments de mot-clé qui doivent être transmis au point de contrôle au cours de la compilation. L'élément `<kwargs>` contient des éléments `<arg name>` qui peuvent être utilisées pour spécifier des mots-clés individuels à transmettre au point de contrôle. L'élément `<arglist>` contient également une liste de plusieurs valeurs `<argitem>` à transmettre au point de contrôle. Cet exemple inclut une liste des fichiers décompressés dans l'élément `<arglist>`.

Chaque élément de liste `<kargs>` est entouré de guillemets. Si aucun guillemet n'est utilisé, ou si un ensemble de guillemets encadre l'intégralité de la chaîne, celle-ci, espaces et retours inclus, est interprétée comme un argument. N'utilisez pas de virgules pour séparer les arguments.

Si vous créez un script personnalisé à utiliser lors de la création d'une image, vous devez ajouter un élément de point de contrôle renvoyant à l'emplacement du script. Le point de contrôle pour un script personnalisé nécessite uniquement un élément `<args>` qui renvoie à l'emplacement du script personnalisé. Pour plus d'informations et d'exemples, reportez-vous à la section [“Création et utilisation de scripts personnalisés” à la page 23](#).

Utilisez les options de la commande `distro_const` pour contrôler la mise en pause et le redémarrage du processus de compilation à des points de contrôle particuliers. Reportez-vous à la section “Création d'une image en plusieurs étapes” à la page 27.

EXEMPLE 2-2 Ajout de packages SVR4

Dans cet exemple, un nouveau point de contrôle est ajouté au fichier manifeste. Ce nouveau point de contrôle répertorie les packages SVR4 à ajouter à l'image et leurs emplacements. Ensuite, ce nouveau point de contrôle est référencé dans la section d'exécution.

Tout d'abord, le nouveau point de contrôle est créé par l'ajout d'un nouvel élément `software`. Ce point de contrôle indique SVR4 comme type de logiciel, l'emplacement des packages et l'emplacement où les installer.

En outre, les packages SVR4 spécifiques à installer sont répertoriés dans l'élément `software_data`.

```
<software name=transfer-svr4-install type="SVR4">
  <destination>
    <dir path={PKG_IMAGE_PATH}/>
  </destination>
  <source>
    <dir path="/path/to/packages"/>
  </source>
  <software_data action="install">
    <name>SUNWpackage1</name>
    <name>SUNWpackage2</name>
  </software_data>
</software>
```

Si les valeurs `{PKG_IMAGE_PATH}` et `{BOOT_ARCHIVE}` sont incluses dans le point de contrôle, l'utilitaire `distro_const` les remplace par `<ZFS Dataset>/build_data/pkg_image` et `<ZFS Dataset>/build_data/boot_archive`, respectivement. Dans cet exemple, les packages SVR4 seront installés dans `<ZFS Dataset>/build_data/pkg_image`.

Enfin, ce nouveau point de contrôle est référencé dans la section d'exécution.

```
<execution stop_on_error="true">
  <checkpoint name="transfer-ips-install"
    desc="Transfer pkg contents from IPS"
    mod_path="solaris_install/transfer/ips"
    checkpoint_class="TransferIPS"/>
  <checkpoint name="set-ips-attributes"
    desc="Set post-install IPS attributes"
    mod_path="solaris_install/transfer/ips"
    checkpoint_class="TransferIPS"/>
  <checkpoint name="transfer-svr4-install"
    desc="Transfer pkg contents from SVR4 packages"
    mod_path="solaris_install/transfer/svr4"
    checkpoint_class="TransferSVR4"/>
```

Notez que le nom du logiciel doit correspondre au nom du point de contrôle. Dans cet exemple, les deux sont "transfer-svr4-install".

Création et utilisation de scripts personnalisés

Le constructeur de distribution vous permet de spécifier des scripts supplémentaires qui peuvent être utilisés afin d'effectuer des personnalisations selon le type d'images créées. Les fichiers manifestes renvoient aux scripts, et ceux-ci transforment l'image générique en une distribution spécifique au média. Ces scripts sont référencés dans la section d'exécution des fichiers manifestes. Vous pouvez spécifier n'importe quel nombre de points de contrôle de scripts personnalisés.

Remarque – La prise en charge des scripts est limitée à tout script par défaut, non modifié, fourni avec les packages d'application. Si vous décidez de personnaliser ces scripts, sauvegardez les scripts d'origine au préalable.

▼ Création et utilisation d'un script personnalisé

Avant de commencer

Lorsque vous créez vos propres scripts personnalisés, prenez en compte les points suivants :

- Les scripts peuvent être des programmes Python, des scripts shell ou des fichiers binaires.
- Les scripts sont exécutés dans leur ordre d'apparition dans la section d'exécution du fichier manifeste.
- La sortie standard (`stdout`) et la sortie d'erreur (`stderr`) des commandes exécutées dans les scripts (modules shell et python) sont consignées dans des fichiers journaux qui établissent des rapports sur les compilations terminées ou tentées.

- 1 **Créez votre nouveau script.**
- 2 **Ajoutez vos nouveaux scripts à votre répertoire personnel ou ailleurs sur le système ou réseau.**
Assurez-vous qu'un utilisateur connecté avec le rôle root peut exécuter ces scripts.
- 3 **Référez le nouveau script en ajoutant un point de contrôle dans la section d'exécution du fichier manifeste approprié.**

Assurez-vous de spécifier le chemin d'accès complet à vos scripts. Les points de contrôle sont exécutés dans leur ordre d'apparition dans la section d'exécution du fichier manifeste.

Lorsque vous ajoutez une référence pour un nouveau script dans la section d'exécution d'un fichier manifeste, vous devez spécifier un nom de point de contrôle pouvant être utilisé pour interrompre la création de l'image avant que ce script ait effectué sa tâche ou après. Vous pouvez également inclure un message personnalisé associé au nom du point de contrôle. Si vous n'ajoutez pas un tel message, le chemin du script est utilisé comme message de point de contrôle par défaut. Un message relatif au point de contrôle s'affiche lorsque le point de contrôle est exécuté au cours du processus de compilation.

Remarque – Attribuez aux points de contrôle des noms explicites plutôt que des nombres. Si de nouveaux scripts sont ajoutés, les nouveaux points de contrôle correspondant à ces nouveaux scripts perturberont l'ordre des points de contrôle numérotés.

L'exemple de point de contrôle suivant fait référence à un script personnalisé nommé "my-script".

```
<checkpoint name="my-script"
  desc="my new script"
  mod_path="solaris_install/distro_const/checkpoints/custom_script"
  checkpoint_class="CustomScript">
  <args>/tmp/myscript.sh</args>
</checkpoint>
```

4 (Facultatif) Spécifiez un paramètre de compilation en tant que partie intégrante du point de contrôle, comme suit.

Ici {PKG_IMAGE_PATH} est spécifié en tant que paramètre de compilation dans la section des arguments.

```
<checkpoint name="my-script"
  desc="my new script"
  mod_path="solaris_install/distro_const/checkpoints/my_script"
  checkpoint_class="CustomScript">
  <args>/tmp/myscript.sh {PKG_IMAGE_PATH}</args>
</checkpoint>
```

Si les valeurs {PKG_IMAGE_PATH} et {BOOT_ARCHIVE} sont incluses dans le point de contrôle, l'utilitaire distro_const les remplace par <ZFS Dataset>/build_data/pkg_image et <ZFS Dataset>/build_data/boot_archive, respectivement.

5 Créez l'image.

Vous pouvez créer une image en une seule étape. Pour vérifier l'état de la compilation, vous pouvez également arrêter et redémarrer la compilation à différents points de contrôle.

Pour obtenir davantage d'instructions, reportez-vous au [Chapitre 3, "Création d'une image"](#).

6 (Facultatif) Une fois la compilation terminée, vous pouvez afficher un fichier journal contenant un rapport sur le processus de compilation.

La sortie de compilation affiche l'emplacement des fichiers journaux.

Création d'une image

Après avoir configuré le fichier manifeste que vous avez choisi d'utiliser et, éventuellement, personnalisé les scripts de finalisation, vous êtes prêt à créer une image en exécutant la commande `distro_const`.

Vous pouvez utiliser la commande `distro_const` pour créer une image de l'une des manières suivantes :

- En une seule étape
- En mettant en pause et en redémarrant la compilation en fonction des besoins afin d'examiner le contenu de l'image et de déboguer les scripts au cours du processus de compilation

Commande `distro_const`

La syntaxe complète pour la commande `distro_const` est la suivante :

Syntax: `distro_const build [-v] [-r checkpoint_name] [-p checkpoint_name] [-l] manifest`

Examinez les options de commande suivantes.

TABLEAU 3-1 Options de la commande `distro_const`

Options de commande	Description
<code>distro_const build <i>manifeste</i></code>	Crée une image en une seule étape à l'aide du fichier manifeste spécifié
<code>distro_const build -v</code>	Mode détaillé
<code>distro_const build -l <i>manifeste</i></code>	Répertorie tous les points de contrôle valides à partir desquels vous pouvez suspendre et reprendre la compilation d'une image

TABLEAU 3-1 Options de la commande distro_const (Suite)	
Options de commande	Description
distro_const build -p <i>nom_point_contrôle</i> <i>manifeste</i>	Suspend la compilation d'une image à un point de contrôle spécifié
distro_const build -r <i>nom_point_contrôle</i> <i>manifeste</i>	Reprend la compilation d'une image à partir d'un point de contrôle spécifié
distro_const build -h	Affiche l'aide de la commande

Remarque – Vous devez vous connecter en tant qu'utilisateur root pour utiliser la commande distro_const.

▼ Création d'une image en une seule étape

Avant de commencer

Téléchargez le package distribution-creator et sélectionnez le fichier manifeste pour votre image. Si nécessaire, personnalisez le fichier manifeste et ajoutez des scripts personnalisés.

- 1 Connectez-vous en tant qu'utilisateur root.
- 2 Pour exécuter la création complète d'une image sans interruption, utilisez la commande **distro_const** de base sans option, comme suit :

```
# distro_const build manifest
```

Remarque – La sous-commande build est requise.

Remplacez *manifeste* par le nom du fichier manifeste à utiliser comme empreinte pour votre image.

Entrez par exemple la commande suivante :

```
# distro_const build /usr/share/distro_const/dc_livecd.xml
```

- 3 Le constructeur de distribution récupère les packages nécessaires pour l'image.
- 4 Le constructeur de distribution crée l'image selon les spécifications définies dans le fichier manifeste.
- 5 (Facultatif) Une fois la compilation terminée, vous pouvez afficher un fichier journal contenant un rapport sur le processus de compilation.

La sortie de compilation affiche l'emplacement des fichiers journaux.

▼ Création d'une image en plusieurs étapes

Vous pouvez utiliser les options offertes par la commande `distro_const` pour arrêter et redémarrer le processus de création à différentes étapes du processus de génération d'image, afin de contrôler et de déboguer votre sélection de fichiers, packages et scripts pour l'image en cours de création. Ce processus utilise les options de création de points de contrôle disponibles dans la commande `distro_const`, comme décrit dans les instructions de base ci-dessous.

- 1 **Connectez-vous en tant qu'utilisateur root.**
- 2 **Avant de créer l'image, vérifiez les points de contrôle valides auxquels vous pouvez choisir de suspendre ou reprendre la compilation.**

```
# distro_const build -l manifest.xml
```

Remarque – La sous-commande `build` est requise.

Cette commande affiche les points de contrôle valides auxquels vous pouvez interrompre ou reprendre la création d'une image. Utilisez les noms de points de contrôle fournis par cette commande comme valeurs correctes pour les autres options de commande de création de points de contrôle.

Par exemple, la commande suivante confirme les points de contrôle disponibles, pour un fichier manifeste nommé `dc_livecd.xml`.

```
# distro_const build -l /usr/share/distro_const/dc_livecd.xml
```

Une fois la commande exécutée, les points de contrôle valides s'affichent. Par exemple, des points de contrôle peuvent inclure les éléments suivants.

Checkpoint	Resumable	Description
transfer-ips-install	X	Transfer package contents from IPS
set-ips-attributes	X	Set post-installation IPS attributes
pre-pkg-img-mod	X	Pre-package image modification
ba-init		Boot archive initialization
ba-config		Boot archive configuration
ba-arch		Boot archive archiving
grub-setup		Set up the GRUB menu
pkg-img-mod		Package image area modifications
create-iso		ISO image creation

Remarque – Dans cet exemple de sortie de commande, une marque X dans le champ de reprise indique que vous pouvez reprendre la création à partir de ce point de contrôle.

- 3 **Créez l'image et mettez la compilation en pause au point de contrôle spécifié.**

```
# distro_const build -p checkpoint_name manifest
```

Remarque – La sous-commande `build`, ainsi que les champs `nom_point_contrôle` et `manifeste` sont obligatoires.

Par exemple, la commande suivante lance la compilation d'une image et l'interrompt avant que `im-mod` ne modifie la zone d'image :

```
# distro_const build -p ba-arch /usr/share/distro_const/dc_livecd.xml
```

4 Reprenez la compilation de l'image à partir d'un point de contrôle spécifié.

```
# distro_const build -r checkpoint_name manifest
```

Remarque – Le point de contrôle spécifié doit être le point de contrôle auquel la compilation antérieure a été interrompue ou un point de contrôle précédent. Un point de contrôle ultérieur n'est pas valide. Les champs `nom_point_contrôle` et `manifeste` et la sous-commande `build` sont requis.

Par exemple, la commande suivante permet de reprendre la compilation de l'image à l'étape `ba-arch`.

```
# distro_const build -r ba-arch /usr/share/distro_const/dc_livecd.xml
```

Remarque – Vous pouvez combiner les options de suspension et de reprise dans une commande `build`.

5 (Facultatif) Une fois la compilation terminée, vous pouvez afficher un fichier journal contenant un rapport sur le processus de compilation.

La sortie de compilation affiche l'emplacement des fichiers journaux.