

Creación de una imagen de instalación personalizada de Oracle® Solaris 11.2



Referencia: E53755
Julio de 2014

Copyright © 2008, 2014, Oracle y/o sus filiales. Todos los derechos reservados.

Este software y la documentación relacionada están sujetos a un contrato de licencia que incluye restricciones de uso y revelación, y se encuentran protegidos por la legislación sobre la propiedad intelectual. A menos que figure explícitamente en el contrato de licencia o esté permitido por la ley, no se podrá utilizar, copiar, reproducir, traducir, emitir, modificar, conceder licencias, transmitir, distribuir, exhibir, representar, publicar ni mostrar ninguna parte, de ninguna forma, por ningún medio. Queda prohibida la ingeniería inversa, desensamblaje o descompilación de este software, excepto en la medida en que sean necesarios para conseguir interoperabilidad según lo especificado por la legislación aplicable.

La información contenida en este documento puede someterse a modificaciones sin previo aviso y no se garantiza que se encuentre exenta de errores. Si detecta algún error, le agradeceremos que nos lo comunique por escrito.

Si este software o la documentación relacionada se entrega al Gobierno de EE.UU. o a cualquier entidad que adquiera licencias en nombre del Gobierno de EE.UU. se aplicará la siguiente disposición:

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

Este software o hardware se ha desarrollado para uso general en diversas aplicaciones de gestión de la información. No se ha diseñado ni está destinado para utilizarse en aplicaciones de riesgo inherente, incluidas las aplicaciones que pueden causar daños personales. Si utiliza este software o hardware en aplicaciones de riesgo, usted será responsable de tomar todas las medidas apropiadas de prevención de fallos, copia de seguridad, redundancia o de cualquier otro tipo para garantizar la seguridad en el uso de este software o hardware. Oracle Corporation y sus filiales declinan toda responsabilidad derivada de los daños causados por el uso de este software o hardware en aplicaciones de riesgo.

Oracle y Java son marcas comerciales registradas de Oracle y/o sus filiales. Todos los demás nombres pueden ser marcas comerciales de sus respectivos propietarios.

Intel e Intel Xeon son marcas comerciales o marcas comerciales registradas de Intel Corporation. Todas las marcas comerciales de SPARC se utilizan con licencia y son marcas comerciales o marcas comerciales registradas de SPARC International, Inc. AMD, Opteron, el logotipo de AMD y el logotipo de AMD Opteron son marcas comerciales o marcas comerciales registradas de Advanced Micro Devices. UNIX es una marca comercial registrada de The Open Group.

Este software o hardware y la documentación pueden ofrecer acceso a contenidos, productos o servicios de terceros o información sobre los mismos. Ni Oracle Corporation ni sus filiales serán responsables de ofrecer cualquier tipo de garantía sobre el contenido, los productos o los servicios de terceros y renuncian explícitamente a ello. Oracle Corporation y sus filiales no se harán responsables de las pérdidas, los costos o los daños en los que se incurra como consecuencia del acceso o el uso de contenidos, productos o servicios de terceros.

Contenido

Uso de esta documentación	5
 1 Introducción a la creación de una imagen de instalación personalizada	7
Acerca del constructor de distribuciones	7
Tipos de imágenes de Oracle Solaris	8
Proceso de creación de imágenes	9
Diferencias de archivos entre SPARC y x86	10
 2 Diseño de una imagen de instalación personalizada	11
Requisitos del sistema para generar imágenes	11
Personalización de imágenes	12
Archivos de manifiesto de ejemplo	12
▼ Cómo crear y generar una imagen personalizada	13
Modificación del contenido del manifiesto	14
Creación y uso de secuencias de comandos personalizadas	24
 3 Generación de una imagen	27
Comando <code>distro_const</code>	27
▼ Cómo generar una imagen en un solo paso	28
▼ Cómo generar una imagen en etapas	28
 Índice	31

Uso de esta documentación

- **Descripción general:** describe cómo crear paquetes de instalación personalizados de Oracle Solaris con la herramienta de construcción de distribuciones.
- **Destinatarios:** técnicos, administradores de sistemas y proveedores de servicios autorizados.
- **Conocimientos previos:** se requiere cierta experiencia con Oracle Solaris.

Biblioteca de documentación del producto

En la biblioteca de documentación (<http://www.oracle.com/pls/topic/lookup?ctx=E56339>), se incluye información de última hora y problemas conocidos para este producto.

Acceso a My Oracle Support

Los clientes de Oracle tienen acceso a soporte electrónico por medio de My Oracle Support. Para obtener más información, visite <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> o, si tiene alguna discapacidad auditiva, visite <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>.

Comentarios

Envíenos comentarios acerca de esta documentación mediante <http://www.oracle.com/goto/docfeedback>.

Introducción a la creación de una imagen de instalación personalizada

Los administradores del sistema y los desarrolladores de aplicaciones pueden utilizar la herramienta de construcción de distribuciones para generar imágenes de instalación personalizadas de Oracle® Solaris.

- Si no ha creado imágenes de instalación personalizadas antes, lea “[Acerca del constructor de distribuciones](#)” [7].
- Si está preparado para generar imágenes personalizadas, vaya a “[Requisitos del sistema para generar imágenes](#)” [11].

Acerca del constructor de distribuciones

El constructor de distribuciones es una herramienta de línea de comandos que permite generar imágenes preconfiguradas de Oracle Solaris. La herramienta utiliza un archivo de manifiesto XML como entrada y genera una imagen basada en los parámetros especificados en el archivo de manifiesto.

El constructor de distribuciones puede generar una imagen ISO, que es un archivo de almacenamiento, también conocida como imagen de disco, de un disco óptico con un formato definido por la Organización Internacional de Normalización (ISO). También puede crear una imagen USB basada en una imagen ISO generada.

Tenga en cuenta los siguientes puntos:

- Según la configuración de imagen, las imágenes ISO o USB pueden ser de inicio.
- Las imágenes ISO y USB se pueden instalar en un sistema o ejecutar en un entorno Live Media.
- Una imagen ISO se puede grabar en un CD o DVD.
- Una imagen USB se puede copiar en una unidad flash.

El constructor de distribuciones crea una imagen USB que funciona en varios tipos de dispositivos de memoria flash que admiten controladores por medio del sistema operativo

Oracle Solaris. Se debe usar la utilidad `usbcopy` para copiar la imagen USB en una unidad flash USB. La utilidad `usbcopy` está disponible en el paquete `distribution-creator`.

Tipos de imágenes de Oracle Solaris

Puede utilizar el constructor de distribuciones para crear los siguientes tipos de imágenes de Oracle Solaris:

- **Live Media x86 de Oracle Solaris.** Puede crear una imagen ISO x86 comparable con la imagen Live Media distribuida con cada versión de Oracle Solaris. También puede personalizar el contenido de esta imagen ISO. Por ejemplo, puede agregar o eliminar paquetes. Puede modificar la configuración predeterminada del entorno de inicio resultante para crear una imagen ISO o USB personalizada.

Para obtener más información sobre las instalaciones Live Media, consulte el [Capítulo 3, “Uso de Live Media”](#) de [“Instalación de sistemas Oracle Solaris 11.2”](#). Para obtener más información sobre cómo personalizar el contenido de la imagen, consulte [“Modificación del contenido del manifiesto”](#) [14].

- **Imagen de instalación de texto x86 o SPARC de Oracle Solaris.** Puede crear una imagen ISO x86 o SPARC que se pueda usar para realizar una instalación de texto del sistema operativo Oracle Solaris. El instalador de texto se puede utilizar en sistemas que no necesitan tarjetas gráficas.

Nota - Una instalación de texto *no* instala todos los paquetes de software que se incluyen al realizar la instalación desde la imagen Live Media. Por ejemplo, el instalador de texto no instala ningún escritorio. Después de una instalación de texto, puede agregar paquetes adicionales, por ejemplo, el paquete `solaris-desktop`.

Para obtener más información sobre las instalaciones de texto, consulte el [Capítulo 4, “Uso del instalador de texto”](#) de [“Instalación de sistemas Oracle Solaris 11.2”](#).

- **Imagen ISO x86 o SPARC para instalaciones automatizadas.** El sistema operativo Oracle Solaris incluye la herramienta Automated Installer. Automated Installer (AI) se utiliza para automatizar la instalación del sistema operativo Oracle Solaris en uno o varios sistemas SPARC y x86 a través de una red. Las instalaciones pueden diferir en la arquitectura, los paquetes instalados, la capacidad de disco y otros parámetros. Puede utilizar el constructor de distribuciones para crear una imagen ISO de AI SPARC que se pueda usar para instalar el sistema operativo Oracle Solaris en clientes SPARC, o bien para crear una imagen ISO de AI x86 que se pueda usar para instalar el sistema operativo Oracle Solaris en clientes x86.

Para obtener información sobre cómo usar Automated Installer, consulte [Parte III, “Instalación con un servidor de instalación”](#) de [“Instalación de sistemas Oracle Solaris 11.2”](#).

Proceso de creación de imágenes

El constructor de distribuciones crea imágenes basadas en la configuración especificada en los archivos XML, denominados *archivos de manifiesto*. Los archivos de manifiesto contienen especificaciones para el contenido y los parámetros de las imágenes ISO creadas con el constructor de distribuciones. El paquete `distribution-creator` contiene manifiestos de ejemplo que se pueden utilizar para crear una imagen ISO personalizada de Live Media x86, una imagen ISO de instalación automatizada x86 o SPARC, o una imagen ISO de instalación de texto x86 o SPARC. Consulte [“Archivos de manifiesto de ejemplo” \[12\]](#).

Todos los campos del archivo de manifiesto proporcionan valores preestablecidos que crearán el tipo de imagen que necesita. Puede editar los campos del archivo de manifiesto para personalizar aún más la imagen resultante. Por ejemplo:

- Puede editar el elemento `target` en el manifiesto para especificar una ubicación diferente para el área de generación donde se puede crear la imagen.
- Puede comprobar el editor especificado y asegurarse de que el sistema que utiliza puede establecer contacto con ese editor para descargar los paquetes necesarios para generar la imagen.
- Puede editar el elemento `software name` para especificar un editor y una ubicación de repositorio diferentes.

Para obtener instrucciones, consulte [“Personalización de imágenes” \[12\]](#).

También puede crear *secuencias de comandos personalizadas* para modificar la imagen de instalación. Luego, puede agregar puntos de comprobación en el archivo de manifiesto para ejecutar estas secuencias de comandos personalizadas. Para obtener más información, consulte [“Creación y uso de secuencias de comandos personalizadas” \[24\]](#).

El paquete `distribution-creator` también incluye una utilidad de línea de comandos, el comando `distro_const`, que interpreta las especificaciones del manifiesto y genera la imagen. Cuando termine de editar el plano de la imagen en un archivo de manifiesto, ejecute el comando `distro_const` para generar la imagen. Para obtener más información, consulte el [Capítulo 3, Generación de una imagen](#).

Puede utilizar las opciones incluidas en el comando `distro_const` para detener y reiniciar la creación en diferentes etapas durante el proceso de generación de imágenes para comprobar y depurar la imagen que se va a crear. Este proceso que consiste en detener y reiniciar el proceso de generación implica la *creación de puntos de comprobación*. La creación de puntos de comprobación es opcional. Los puntos de comprobación predeterminados se especifican en cada archivo de manifiesto.

Tras ejecutar el comando `distro_const`, puede consultar el archivo log simple o detallado para ver la información relacionada con la generación.

Para obtener más información, consulte [Cómo generar una imagen en etapas \[28\]](#) o la página del comando `man distro_const(1M)`.

Diferencias de archivos entre SPARC y x86

El archivo raíz de las imágenes x86 es diferente del archivo raíz de las imágenes SPARC. El archivo raíz completo, o `boot_archive`, de las imágenes x86 es un sistema de archivos UFS, comprimido con `lzma`. La plataforma SPARC no admite la compresión del archivo raíz completo de esta forma. En su lugar, los archivos raíz SPARC usan DCFS, que comprime cada archivo de forma individual. Estos archivos comprimidos individualmente pueden requerir un tratamiento específico en el manifiesto. Para obtener instrucciones, consulte el campo `<boot_archive_contents>` en la página del comando `man dc_manifest(4)`.

♦ ♦ ♦ 2 CAPÍTULO 2

Diseño de una imagen de instalación personalizada

En este capítulo, se proporcionan los requisitos del sistema y se explica cómo diseñar una imagen de instalación personalizada.

Requisitos del sistema para generar imágenes

Para poder utilizar el constructor de distribuciones, debe cumplir los requisitos del sistema que se describen en la tabla siguiente.

TABLA 2-1 Requisitos del sistema

Requisito	Descripción
Espacio en disco	El tamaño mínimo recomendado para el espacio de trabajo del constructor de distribuciones es de 8 GB. Confirme que tiene espacio suficiente en el sistema para utilizar el constructor de distribuciones.
Sistema operativo Oracle Solaris	Debe tener el sistema operativo Oracle Solaris instalado en el sistema. Tenga en cuenta las siguientes consideraciones: ■ El sistema instalado debe tener acceso a la red. El constructor de distribuciones accede a los repositorios Image Packaging System (IPS) que están disponibles en la red para recuperar paquetes para la imagen ISO. Debe tener acceso de red a los repositorios que se especifiquen en el archivo de manifiesto. ■ Al utilizar el constructor de distribuciones, puede crear sólo imágenes SPARC en un sistema SPARC y sólo imágenes x86 en un sistema x86. ■ La versión de Oracle Solaris del sistema debe ser igual a la versión de la imagen que desea crear con el constructor de distribuciones. Nota - Para ejecutar el constructor de distribuciones, debe asumir el rol root.
Paquetes necesarios	El paquete <code>distribution-creator</code> , que contiene la herramienta de construcción de distribuciones.

Personalización de imágenes

El constructor de distribuciones crea imágenes basadas en la configuración especificada en los archivos XML, denominados *archivos de manifiesto*. Los archivos de manifiesto contienen especificaciones para el contenido y los parámetros de las imágenes ISO creadas con el constructor de distribuciones. El paquete `distribution-creator` contiene manifiestos de ejemplo que se pueden utilizar para crear una imagen ISO personalizada de Live Media x86, una imagen ISO de instalación automatizada x86 o SPARC, o una imagen ISO de instalación de texto x86 o SPARC.

Los elementos de cada archivo de manifiesto proporcionan valores preestablecidos que crearán el tipo de imagen ISO que necesita. Puede editar manualmente estos elementos preestablecidos en un archivo de manifiesto para personalizar la imagen resultante. Además, puede crear secuencias de comandos personalizadas para modificar aún más la imagen. A continuación, haga referencia a las nuevas secuencias de comandos en el archivo de manifiesto.

Archivos de manifiesto de ejemplo

El paquete `distribution-creator` proporciona los archivos de manifiesto de ejemplo que se describen en la siguiente tabla.

TABLA 2-2 Manifiestos de ejemplo

Tipo de manifiesto	Ubicación de manifiesto	Descripción
Imagen ISO de Live Media x86	<code>/usr/share/distro_const/ dc_livecd.xml</code>	Se utiliza para crear una imagen ISO x86 comparable con la imagen Live Media de Oracle Solaris.
Imagen de instalación de texto x86	<code>/usr/share/distro_const/ dc_text_x86.xml</code>	Se utiliza para crear una imagen ISO x86 que se puede usar para realizar una instalación de texto del sistema operativo Oracle Solaris en clientes x86.
Imagen de instalación de texto SPARC	<code>/usr/share/distro_const/ dc_text_sparc.xml</code>	Se utiliza para crear una imagen ISO SPARC que se puede usar para realizar una instalación de texto del sistema operativo Oracle Solaris en clientes SPARC.
Imagen ISO de AI x86	<code>/usr/share/distro_const/ dc_ai_x86.xml</code>	Se utiliza para crear una imagen ISO de instalación automatizada x86 para instalaciones automatizadas del sistema operativo Oracle Solaris en clientes x86.
Imagen ISO de AI SPARC	<code>/usr/share/distro_const/ dc_ai_sparc.xml</code>	Se utiliza para crear una imagen ISO de instalación automatizada SPARC para instalaciones automatizadas del sistema operativo Oracle Solaris en clientes SPARC.

▼ Cómo crear y generar una imagen personalizada

En este procedimiento, se describen los pasos generales para crear y generar una imagen personalizada.

1. Conviértase en administrador.

Para obtener más información, consulte [“Uso de sus derechos administrativos asignados” de “Protección de los usuarios y los procesos en Oracle Solaris 11.2”](#).

2. Instale el paquete `distribution-creator`, que contiene la herramienta de construcción de distribuciones y los manifiestos de ejemplo.

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

3. Copie uno de los manifiestos de ejemplo y cree un archivo de manifiesto personalizado con un nuevo nombre de archivo.

Hará referencia al archivo de manifiesto por el nombre cuando use el comando `distro_const` para crear una imagen.

Nota - Realice siempre una copia de seguridad del archivo de manifiesto original y las secuencias de comandos predeterminadas antes de copiarlos.

4. Edite los elementos del manifiesto.

Por ejemplo, puede editar el elemento `target` en el manifiesto para especificar una ubicación diferente para el área de generación donde se puede crear la imagen. También puede comprobar el editor para asegurarse de que el sistema puede establecer contacto con ese editor para descargar los paquetes necesarios para generar la imagen. Si es necesario, puede editar el elemento `software name` para especificar un editor y una ubicación de repositorio diferentes.

Para obtener información, consulte [“Modificación del contenido del manifiesto” \[14\]](#) y la página del comando `man dc_manifest(4)`.

5. (Opcional) Cree secuencias de comandos personalizadas para modificar aún más la imagen.

Si crea nuevas secuencias de comandos, actualice sus referencias en la sección de ejecución del archivo de manifiesto.

Para obtener instrucciones, consulte [“Creación y uso de secuencias de comandos personalizadas” \[24\]](#).

6. Ejecute la utilidad `distro_const` para crear una imagen.

Para obtener instrucciones, consulte el [Capítulo 3, Generación de una imagen](#).

Modificación del contenido del manifiesto

Todos los campos del archivo de manifiesto proporcionan valores preestablecidos que crearán el tipo de imagen ISO que necesita. Puede editar manualmente estos campos preestablecidos en un archivo de manifiesto para personalizar aún más la imagen resultante.

En la tabla siguiente, se describen los principales elementos de los archivos de manifiesto de ejemplo.

TABLA 2-3 Elementos de manifiesto

Elemento	Descripción
<code><distro name="Oracle_Solaris_Text_X86" add_timestamp="false"></code>	Especifica el nombre de la imagen con un indicador de fecha y hora opcional.
<code><boot_mods></code>	Especifica las modificaciones del menú de GRUB para la imagen.
<code><target></code>	Define el conjunto de datos de la agrupación ZFS donde se genera la imagen.
<code><software name="transfer-ips-install" type="IPS"></code>	Especifica el origen de los paquetes de software que se instalarán.
<code><software_data action="install"></code>	Enumera los paquetes que se instalarán.
<code><software_data action="uninstall"></code>	Enumera los paquetes que se desinstalarán.
<code><software name="set-ips-attributes"></code>	Establece diferentes atributos para IPS una vez finalizada la instalación.
<code><software name="ba-init"></code>	Especifica el contenido del archivo de inicio. Atención - Se debe modificar con precaución. Si el archivo de inicio es incorrecto, el sistema instalado no se podrá iniciar.
<code><execution stop_on_error="true"></code> <code><checkpoint name="transfer-ips-install"/></code>	Enumera los puntos de comprobación de la generación.
<code><configuration name="pre-pkg-img-mod" type="sysconf"</code> <code>source="/etc/svc/profile/generic_limited_net.xml"></code>	Especifica los servicios SMF que se aplicarán a los medios durante la generación. Atención - Casi nunca se debe modificar.

Especificación del título de la imagen

Utilice el siguiente elemento para proporcionar un nombre predeterminado o personalizado para la imagen que desea generar:

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

Si desea generar varias versiones de una imagen y conservar las imágenes incrementales, puede cambiar la variable timestamp a "true", y se anejará automáticamente un indicador de fecha y hora al nombre de cada imagen.

Si necesita especificar un proxy HTTP, quite el comentario del elemento `distro name` que incluye la variable `proxy` y proporcione la ubicación del proxy.

Modificación del menú de inicio

Este elemento de menú de inicio especifica las modificaciones del menú de inicio que se aplicarán a la imagen.

En el ejemplo siguiente, se aplicará a la imagen un menú de inicio especializado con el título “boot1”. El atributo `timeout` especifica el tiempo que transcurre antes de que se active automáticamente la entrada de inicio predeterminada.

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

Dentro del elemento de menú de inicio, puede agregar entradas de menú de inicio individuales. Para ello, agregue un elemento `boot_entry` nuevo para cada entrada nueva. Se agregan entradas de forma secuencial al menú de inicio en el orden determinado por el valor de atributo `insert_at` de “inicio” o “fin” para cada entrada de inicio.

Nota - Agregue entradas nuevas antes de la entrada “con magnificador” existente.

Vea el siguiente ejemplo de un elemento `boot_entry` individual.

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

Para obtener información detallada, consulte la página del comando `man dc_manifest(4)`.

Especificación del área de generación

Puede personalizar el elemento `target`. Este elemento define el conjunto de datos de la agrupación ZFS que se utilizará para la generación. Este conjunto de datos representa el área donde se creará la imagen. Debe proporcionar una ubicación válida para el conjunto de datos. Debe comprobar el área de generación predeterminada para asegurarse de que la creación no destruirá el contenido que necesita conservar en el sistema. Modifique el área de generación si es necesario.

Nota - El nombre del sistema de archivos no debe incluir el nombre de `zpool`.

En el siguiente ejemplo, se muestra un elemento `target`.

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

Especificación del editor

El siguiente elemento especifica un editor en el que el constructor de distribuciones puede obtener paquetes para descargar y usar para generar la imagen.

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

En el elemento source anidado en la sección de nombre de software, edite los elementos publisher name y origin name para especificar qué editor se debe utilizar y dónde se encuentra el repositorio de paquetes. La ubicación del repositorio puede ser una ruta NFS o un directorio local. Se pueden enumerar varios editores. Cuando el constructor de distribuciones intenta ubicar paquetes para instalar, la búsqueda en los editores se realiza en el orden en que se muestran aquí.

Si se deben especificar reflejos para un editor, quite el comentario y edite el elemento mirror name.

En el ejemplo siguiente, se muestra un elemento source detectado en el elemento software name.

```
<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"/>
  </publisher>
  <publisher name="publisher3.org">
    <origin name="http://example3.com/dev"/>
  </publisher>
  <publisher name="publisher4">
    <origin name="file:///net/myserver/publisher4/repo"/>
  </publisher>
</source>
```

Para obtener más información sobre el uso de los editores, consulte [“Agregación y actualización de software en Oracle Solaris 11.2”](#).

Enumeración de los paquetes para instalar

El elemento `software_data` con el atributo `install` muestra el conjunto de paquetes que instalará para crear un determinado tipo de imagen, según el manifiesto que se utiliza. Por ejemplo, el manifiesto `dc_livecd.xml` muestra los paquetes necesarios para generar una imagen Live Media. Cada etiqueta `name` muestra el nombre de un paquete o el nombre de un paquete de grupo que contiene muchos paquetes.

```
<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 tiene paquetes que desea agregar a la imagen, agregue una etiqueta `name` a cada paquete para anexar los nombres de paquetes.

De forma predeterminada, se instala la última versión de paquete disponible en el repositorio especificado. Si se necesita otra versión, anexe el número de versión a la referencia de paquete con el siguiente formato:

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

Nota - La versión de Oracle Solaris del sistema debe ser igual a la versión de la imagen que desea crear con el constructor de distribuciones.

Además, es posible que no se instalen paquetes con una determinada versión si se instalan otros paquetes con una versión en conflicto según se especifica en el archivo de manifiesto de un servicio de instalación automatizada. Consulte el [Capítulo 9, “Personalización de instalaciones de “Instalación de sistemas Oracle Solaris 11.2”](#)”.

EJEMPLO 2-1 Agregación de paquetes y otros editores

En este ejemplo, se especifica un segundo editor, `mypublisher`. Se especifican paquetes adicionales, `mypackage1` y `mypackage2`.

Durante el proceso de generación, los editores se comprueban en el orden en que se muestran. Si no se encuentran paquetes en el primer editor, los paquetes especificados se buscan en el siguiente editor.

```
<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">
  <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>
```

Enumeración de los paquetes para desinstalar

El elemento `software_data` con el atributo `uninstall` se puede utilizar para desinstalar un paquete individual o para desinstalar una definición de paquete de grupo.

Nota - Una **definición de paquete de grupo** enlaza todos los paquetes individuales dentro de ese grupo en una unidad en la que sólo se puede actuar como grupo.

El atributo `uninstall` resulta especialmente útil si desea instalar un paquete de grupo completo, pero desea omitir uno o más paquetes individuales de dicho grupo. Puede utilizar el atributo `uninstall` para eliminar primero la definición de paquete de grupo. A continuación, puede desinstalar paquetes individuales que se instalaron como parte de un paquete de grupo.

Por ejemplo, es posible que haya decidido generar una imagen de instalación Live Media. La imagen de instalación Live Media predeterminada incluye un explorador Firefox en el paquete de grupo del escritorio.

Si desea omitir el explorador Firefox de la imagen que desea generar, debe hacer lo siguiente:

1. Instale el paquete de grupo `solaris-desktop` que incluye todo el software para el escritorio Live Media habitual. Consulte [“Enumeración de los paquetes para instalar”](#) [17].
2. Desinstale la definición de paquete de grupo `solaris-desktop` con el atributo `uninstall`, de la siguiente manera:

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

Nota - La acción de desinstalación en el paquete de grupo sólo desinstala la definición de paquete de grupo. Todos los paquetes individuales dentro de ese grupo permanecen instalados como se especificó en el primer paso.

3. Ahora que los paquetes individuales no están enlazados en una definición de grupo, puede volver a utilizar el atributo `uninstall` para desinstalar el paquete de Firefox.

```
<software_data action="uninstall">  
  <name>pkg:/web/browser/firefox</name>  
</software_data>
```

También puede combinar los pasos 2 y 3 en una entrada, de la siguiente forma:

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

Anexe paquetes adicionales para desinstalar al final de la sección de desinstalación.

Especificación del editor para un sistema instalado

El elemento `software name` afecta un sistema una vez que ese sistema se instaló con la imagen creada mediante el constructor de distribuciones.

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

Proporcione la etiqueta `publisher name` y la etiqueta opcional `mirror name` para especificar dónde puede obtener el sistema instalado otros paquetes para descargar e instalar.

También puede establecer atributos IPS en este elemento. Consulte información sobre la propiedad IPS en la página del comando `man pkg(1)`.

Configuración de puntos de comprobación de la generación

El elemento `execution` del manifiesto muestra una serie de puntos de comprobación que se ejecutan durante el proceso de creación de imágenes. Los puntos de comprobación se ejecutan en el orden en que se muestran en esta sección. Los puntos de comprobación predeterminados necesarios para generar la imagen de instalación predeterminada se incluyen en cada manifiesto.

Durante el proceso de creación de imágenes, los puntos de comprobación modifican el contenido del área de generación especificada en el manifiesto.

El área de generación contiene los siguientes directorios:

- `ZFS dataset/build_data/pkg_image`

- *ZFS dataset/build_data/boot_archive*

Donde la variable *ZFS dataset* está especificada por el elemento target en el manifiesto.

Durante el proceso de generación, todos los elementos que se incluirán en la imagen final se agregan al directorio *pkg_image*. Los archivos del directorio *boot_archive* independiente se utilizan durante el proceso de generación para crear un archivo de almacenamiento de inicio que también se agrega al directorio *pkg_image*.

En la siguiente lista, se ofrece una breve descripción de cada punto de comprobación predeterminado en el orden en que se ejecutan en la mayoría de los manifiestos.

- *transfer-ips-install*: en este punto de comprobación, el constructor de distribuciones establece contacto con los editores IPS y agrega a la imagen los paquetes que se enumeran en el elemento *software_data* del manifiesto.
- *set-ips-attributes*: en este punto de comprobación, el constructor establece el editor que utilizará el sistema instalado. Los valores establecidos por este punto de comprobación no son relevantes si desea generar una imagen de instalación automatizada.
- *pre-pkg-img-mod*: en este punto de comprobación, el constructor importa en la imagen los archivos del servicio SMF que se especificaron en el elemento *configuration* del manifiesto. Además, el constructor modifica algunos archivos para optimizar la imagen. Todos los cambios realizados hasta este punto de comprobación se incluyen en la imagen que se desea generar y en el archivo raíz. Debe agregar nuevos puntos de comprobación para las secuencias de comandos personalizadas antes o inmediatamente después del punto de comprobación *pre-pkg-img-mod* si desea asegurarse de que los cambios de las secuencias de comandos personalizadas se incorporen en el archivo raíz y en la imagen.
- *ba-init*: en este punto de comprobación, el constructor rellena el archivo raíz con los archivos enumerados en la sección *ba-init* del manifiesto. Estos archivos se copian del área *pkg_image* al área *root_archive*.
- *ba-config*: en este punto de comprobación, el constructor realiza modificaciones adicionales en los archivos copiados al archivo raíz. El constructor crea enlaces simbólicos a otros archivos que no se necesitan hasta más adelante en el proceso de inicio a fin de minimizar el tamaño del archivo raíz.
- *ba-arch*: en este punto de comprobación, el constructor empaqueta el archivo raíz y crea el archivo raíz como un archivo dentro del directorio *pkg_image*. El constructor también aplica todas las optimizaciones al archivo raíz que son específicas del tipo de sistema que se desea crear. Después de este punto de comprobación, los cambios en las especificaciones del archivo de inicio realizados por secuencias de comandos personalizadas no se integrarán en el archivo raíz porque éste ya se empaquetó.
- *grub-setup*: en este punto de comprobación, el constructor configura el menú de GRUB2 según las entradas especificadas en la sección *boot_entry* del manifiesto. Este punto de comprobación sólo se aplica a las imágenes de los sistemas x86.
- *pkg-img-mod*: en este punto de comprobación, el constructor crea los principales archivos de la imagen que se desea generar y optimiza el área *pkg_image*. El constructor mueve los

archivos del directorio `pkg_image` y crea el archivo de la imagen. Todos los elementos del directorio `pkg_image` se incluyen en la imagen. Las adiciones posteriores a este punto de comprobación no se incluirán en la imagen.

- `create-iso`: en este punto de comprobación, se generan los archivos `.iso`, incluidos todos los elementos del directorio `pkg_image`.
- `create-usb`: en este punto de comprobación, se genera el archivo `.usb` a partir del archivo `.iso` generado.

Al observar los campos específicos incluidos en las secciones de punto de comprobación, cada etiqueta `checkpoint name` incluye el atributo `mod-path` que especifica dónde se encuentra la secuencia de comandos del punto de comprobación.

Algunas de las etiquetas `checkpoint` predeterminadas incluyen argumentos con valores predeterminados proporcionados. En el siguiente punto de comprobación de ejemplo del manifiesto `dc_ai_sparc.xml`, se crea un archivo de inicio para la generación de imágenes y se hace referencia a una secuencia de comandos que llevará a cabo esa tarea. En el punto de comprobación de ejemplo, también se incluyen campos de argumentos con valores específicos proporcionados para cada uno de ellos.

```
<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>
```

Como se muestra en este ejemplo, el elemento `kwargs` contiene argumentos de palabras clave que se deben transferir al punto de comprobación durante el proceso de generación. Dentro del elemento `kwargs`, hay elementos `arg name` que se pueden utilizar para especificar palabras clave individuales que se transferirán al punto de comprobación. Además, el elemento `arglist` contiene una lista de varios valores `argitem` que se transferirán al punto de comprobación. En este ejemplo, se incluye una lista de archivos no comprimidos en el elemento `arglist`.

Todos los elementos de lista `kargs` se escriben entre comillas dobles. Cuando no se utilizan comillas dobles o si un juego de comillas dobles abarca la cadena completa, entonces la cadena completa, incluidos los espacios y las líneas nuevas, se interpreta como un argumento. No utilice comas entre los argumentos.

Si crea una secuencia de comandos personalizada para usar durante la generación de una imagen, debe agregar un elemento checkpoint que haga referencia a la ubicación de la secuencia de comandos. El punto de comprobación de una secuencia de comandos personalizada sólo necesita un elemento `args` que haga referencia a la ubicación de la secuencia de comandos personalizada. Para obtener información adicional y ejemplos, consulte [“Creación y uso de secuencias de comandos personalizadas” \[24\]](#).

Utilice las opciones del comando `distro_const` para controlar cómo detener y reiniciar el proceso de generación en puntos de comprobación determinados. Consulte [Cómo generar una imagen en etapas \[28\]](#).

EJEMPLO 2-2 Agregación de paquetes SVR4

En este ejemplo, se agrega un nuevo punto de comprobación al manifiesto. Este nuevo punto de comprobación muestra los paquetes SVR4 que se agregarán a la imagen y su ubicación. Luego, se hace referencia a este nuevo punto de comprobación en la sección de ejecución.

En primer lugar, se agrega un nuevo elemento `software` para crear el nuevo punto de comprobación. Este punto de comprobación especifica SVR4 como tipo de software, dónde se encuentran los paquetes y dónde se instalarán.

Además, los paquetes SVR4 específicos que se instalarán se enumeran en el elemento `software_data`.

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

Si se incluyen en el punto de comprobación, los valores de `{PKG_IMAGE_PATH}` y `{BOOT_ARCHIVE}` son reemplazados por la utilidad `distro_const` por `ZFS dataset / build_data/pkg_image` y `ZFS dataset / build_data/boot_archive`, respectivamente. En este ejemplo, los paquetes SVR4 se instalarán en `ZFS dataset / build_data/pkg_image`.

Por último, se hace referencia al nuevo punto de comprobación en la sección de ejecución. La etiqueta checkpoint name puede ser cualquier cadena, pero para este ejemplo, checkpoint_class debe ser TransferSVR4.

```
<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"/>
```

Tenga en cuenta que los elementos software name y checkpoint name deben coincidir. En este ejemplo, ambos son “transfer-svr4-install”.

EJEMPLO 2-3 Creación de hashes de los medios

Este punto de comprobación permite a los usuarios generar automáticamente hashes de los medios generados por distro_const mediante el punto de comprobación de sumas de comprobación en el manifiesto DC.

```
<checkpoint name="checksums"
  desc="Checksum calculation for media"
  mod_path="solaris_install/distro_const/checkpoints/checksums"
  checkpoint_class="Checksums">
  <kwargs>
    <arglist name="algorithms">
      <argitem file_path="/tmp/md5sums.txt">md5</argitem>
      <argitem>sha1</argitem>
      <argitem>sha224</argitem>
      <argitem>sha256</argitem>
      <argitem>sha384</argitem>
      <argitem>sha512</argitem>
    </arglist>
  </kwargs>
</checkpoint>
```

El elemento arglist incluye todos los algoritmos que se utilizan para generar los hashes de los medios generados. Cada argitem especifica un algoritmo. Los algoritmos válidos se pueden determinar mediante la ejecución de `/usr/bindigest -l`. Cada argitem puede tener un atributo path que especifique la ruta absoluta de un archivo adicional que se agregará con los hashes producidos por ese algoritmo. Si no se especifican algoritmos, el valor predeterminado es md5.

Mientras se genera la imagen, se generan archivos para cada algoritmo que contienen sumas de comprobación para cada medio.

Creación y uso de secuencias de comandos personalizadas

El constructor de distribuciones permite especificar secuencias de comandos adicionales que se pueden utilizar para hacer personalizaciones durante el proceso de creación de imágenes en función del tipo de imagen que se genera. Los archivos de manifiesto hacen referencia a las secuencias de comandos, y las secuencias de comandos transforman la imagen genérica en una distribución específica del medio. Se hace referencia a estas secuencias de comandos en la sección de ejecución de los archivos de manifiesto. Se puede especificar cualquier cantidad de puntos de comprobación de secuencias de comandos personalizadas.

Nota - La compatibilidad con secuencias de comandos está limitada a las secuencias de comandos predeterminadas sin modificar que se proporcionan con los paquetes de aplicaciones. Si decide personalizar estas secuencias de comandos, realice primero una copia de seguridad de las secuencias de comandos originales.

Además, tenga en cuenta que las secuencias de comandos especificadas en la sección de ejecución del archivo de manifiesto se ejecutan durante el proceso de creación de imágenes. La sección de ejecución no hace referencia a secuencias de comandos anteriores o posteriores a la instalación.

Al crear secuencias de comandos personalizadas, tenga en cuenta lo siguiente:

- Las secuencias de comandos pueden ser programas Python, secuencias de comandos de shell o archivos binarios.
- Las secuencias de comandos se ejecutan en el orden en que se muestran en la sección de ejecución del archivo de manifiesto.
- La salida estándar (`stdout`) y la salida de error (`stderr`) de los comandos ejecutados dentro de las secuencias de comandos (módulos de shell y python) se capturan en archivos de registro que informan sobre las generaciones que se completaron o que se intentaron.

▼ Cómo crear y usar una secuencia de comandos personalizada

1. Cree la nueva secuencia de comandos.
2. Agregue las nuevas secuencias de comandos en el directorio raíz o en cualquier otra ubicación de la red o del sistema.

Asegúrese de que el usuario que asuma el rol root pueda ejecutar estas secuencias de comandos.

3. Haga referencia a la nueva secuencia de comandos. Para ello, agregue un punto de comprobación en la sección de ejecución del archivo de manifiesto correspondiente.

Para decidir dónde desea configurar un nuevo punto de comprobación, consulte las descripciones de los puntos de comprobación predeterminados como se describen en [“Configuración de puntos de comprobación de la generación” \[19\]](#).

Asegúrese de especificar la ruta completa de las secuencias de comandos. Los puntos de comprobación se ejecutan en el orden en que se muestran en la sección de ejecución del manifiesto.

Cuando agrega una referencia para una nueva secuencia de comandos en la sección de ejecución de un archivo de manifiesto, debe especificar un nombre de punto de comprobación que se pueda utilizar para pausar la generación de imágenes antes o después de que esta secuencia de comandos realice su tarea. También puede incluir un mensaje personalizado asociado al nombre de punto de comprobación. Si este mensaje se omite, la ruta de la secuencia de comandos se utiliza como mensaje de punto de comprobación predeterminado. El mensaje de punto de comprobación aparece cuando se ejecuta el punto de comprobación durante el proceso de generación.

Nota - Utilice nombres descriptivos para los puntos de comprobación en lugar de usar números. Si se agregan nuevas secuencias de comandos, los nuevos puntos de comprobación de esas nuevas secuencias de comandos alterarán el orden numerado.

En el ejemplo siguiente, el punto de comprobación hace referencia a una secuencia de comandos personalizada denominada “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. (Opcional) Especifique un parámetro de generación como parte del punto de comprobación de la siguiente manera.

Aquí {PKG_IMAGE_PATH} se especifica como parámetro de creación en la sección de argumentos.

```
<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 se incluyen en el punto de comprobación, los valores de {PKG_IMAGE_PATH} y {BOOT_ARCHIVE} son reemplazados por la utilidad `distro_const` por *ZFS dataset / build_data/pkg_image* y *ZFS dataset/build_data/boot_archive*, respectivamente.

5. Genere la imagen.

Puede generar la imagen en un solo paso. O bien, para comprobar el estado de la generación, puede detener y reiniciar la creación en distintos puntos de comprobación.

Para obtener instrucciones, consulte el [Capítulo 3, Generación de una imagen](#).

6. (Opcional) Una vez finalizada la creación, puede ver un archivo de registro con información sobre el proceso de generación.

La salida de generación muestra la ubicación de los archivos de registro.

Generación de una imagen

Una vez que ha configurado el archivo de manifiesto que planea usar y, si lo desea, personalizado las secuencias de comandos de finalización, ya está listo para generar una imagen mediante la ejecución del comando `distro_const`.

Puede utilizar el comando `distro_const` para generar una imagen en un solo paso. O bien, puede pausar y reiniciar la creación según sea necesario para examinar el contenido de la imagen y depurar las secuencias de comandos durante el proceso de generación.

Comando `distro_const`

La sintaxis completa del comando `distro_const` es la siguiente:

```
distro_const build [-v] [-r checkpoint] [-p checkpoint] [-l] manifest
```

Las opciones del comando `distro_const` se describen en la tabla siguiente.

TABLA 3-1 Opciones del comando `distro_const`

Opciones de comando	Descripción
<code>distro_const build manifest</code>	Genera una imagen en un solo paso mediante el archivo de manifiesto especificado.
<code>distro_const build -v manifest</code>	Modo detallado.
<code>distro_const build -l manifest</code>	Muestra todos los puntos de comprobación válidos en los que puede pausar y reanudar la generación de una imagen.
<code>distro_const build -p checkpoint manifest</code>	Pausa la generación de una imagen en un punto de comprobación especificado.
<code>distro_const build -r checkpoint manifest</code>	Reanuda la generación de una imagen desde un punto de comprobación especificado.
<code>distro_const build -h</code>	Muestra la ayuda del comando.

Nota - Debe asumir el rol de usuario root para utilizar el comando `distro_const`.

▼ Cómo generar una imagen en un solo paso

1. **Conviértase en administrador.**

Para obtener más información, consulte [“Uso de sus derechos administrativos asignados” de “Protección de los usuarios y los procesos en Oracle Solaris 11.2”](#).

2. **Descargue el paquete `distribution-creator`.**

3. **Seleccione un manifiesto para la imagen.**

4. **(Opcional) Si es necesario, personalice el manifiesto. Para ello, agregue referencias a las secuencias de comandos personalizadas.**

5. **Ejecute el comando `distro_const` básico sin opciones.**

```
# distro_const build manifest.xml
```

Reemplace *manifest* por el nombre del archivo de manifiesto que se utilizará como plano de la imagen.

Por ejemplo:

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

El constructor de distribuciones extrae los paquetes necesarios para la imagen y genera la imagen según las especificaciones definidas en el archivo de manifiesto.

6. **(Opcional) Una vez finalizada la creación, puede ver un archivo de registro con información sobre el proceso de generación.**

La salida de generación muestra la ubicación de los archivos de registro.

▼ Cómo generar una imagen en etapas

Puede utilizar las opciones incluidas en el comando `distro_const` para detener y reiniciar la creación en diferentes puntos de comprobación durante el proceso de generación de imágenes, a fin de comprobar y depurar la selección de archivos, paquetes y secuencias de comandos para la imagen que se va a crear.

1. **Conviértase en administrador.**

Para obtener más información, consulte [“Uso de sus derechos administrativos asignados” de “Protección de los usuarios y los procesos en Oracle Solaris 11.2”](#).

2. **Descargue el paquete `distribution-creator`.**

3. **Seleccione el manifiesto de la imagen.**
4. **(Opcional) Si es necesario, personalice el manifiesto. Para ello, agregue referencias a las secuencias de comandos personalizadas.**
5. **Revise los puntos de comprobación válidos en los que puede elegir pausar o reanudar la generación.**

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

Este comando muestra los puntos de comprobación válidos en los que puede pausar o reanudar la generación de una imagen. Utilice los nombres de puntos de comprobación proporcionados por este comando como valores válidos para las demás opciones de comandos de creación de puntos de comprobación.

Por ejemplo, el siguiente comando confirma qué puntos de comprobación están disponibles para un archivo de manifiesto denominado `dc_livecd.xml`.

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

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	X Boot archive initialization
ba-config	X Boot archive configuration
ba-arch	X Boot archive archiving
boot-setup	Setup LiveCD boot menu
pkg-img-mod	Package image area modifications
create-iso	ISO image creation
create-usb	USB image creation

Nota - En esta salida del comando de ejemplo, una “X” en el campo resumable indica que puede reiniciar la generación desde este punto de comprobación.

6. **Genere la imagen y pause la creación de la imagen en el punto de comprobación especificado.**

```
# distro_const build -p checkpoint manifest
```

Por ejemplo, el siguiente comando inicia la creación de una imagen y la detiene antes de que `ba-arch` modifique el área de la imagen:

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

7. **Reanude la generación de la imagen desde un punto de comprobación especificado.**

```
# distro_const build -r checkpoint manifest
```

Nota - El punto de comprobación especificado debe ser el punto de comprobación en el que se detuvo la ejecución de la creación anterior o un punto de comprobación previo. No es válido usar un punto de comprobación posterior.

Por ejemplo, el siguiente comando reanuda la generación de la imagen en la etapa `ba-arch`.

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

Nota - Puede combinar las opciones de pausa y reanudación en un comando `build`.

8. (Opcional) Una vez finalizada la creación, puede ver un archivo de registro con información sobre el proceso de generación.

La salida de generación muestra la ubicación de los archivos de registro.

Índice

A

- archivo raíz
 - diferencias entre SPARC y x86, 10
- archivos de manifiesto
 - definición de, 9
 - ejemplo, 12
 - personalización, 14
- archivos de manifiesto de ejemplo, 12
- asignación de nombre
 - imagen de instalación, 14
 - puntos de comprobación, 25

C

- comando `distro_const`
 - sintaxis y opciones, 27
 - uso para generar una imagen en etapas, 28
- constructor de distribuciones
 - descripción general, 7

D

- desinstalación de paquetes, 18
- dispositivos de memoria flash, imágenes de instalación USB y, 7

E

- editor
 - agregación en imagen de instalación, 17
 - especificación para paquetes, 16
 - modificación para sistema instalado, 19
- elemento de manifiesto `boot_entry`, 15
- elemento de manifiesto `boot_mods`, 15
- elemento de manifiesto `distro name`, 14

- elemento de manifiesto `execution`, 19
- elemento de manifiesto `source`, 16
- elemento de manifiesto `target`, 15
- elementos de manifiesto
 - `boot_entry`, 15
 - `boot_mods`, 15
 - `distro name`, 14
 - `execution`, 19
 - lista de, 14
 - modificación
 - archivo raíz, 10
 - área de generación, 15
 - editor para sistema instalado, 19
 - especificación de editor para usar durante la generación, 16
 - lista de paquetes, 17
 - menú de inicio, 15
 - puntos de comprobación de generación, 19
 - título de imagen, 14
 - `source`, 16
 - `target`, 15

G

- generación de imágenes de instalación, 13
 - descripción general, 27
 - en un solo paso, 28
 - imágenes de instalación en etapas, 28

I

- imágenes de instalación
 - agregación de otros editores en, 17
 - agregación de paquetes en, 17
 - agregación de paquetes SVR4 en, 22

- asignación de nombre, 14
- generación, 13, 27
 - descripción general, 9
 - en etapas, 28
 - en un solo paso, 28
 - requisitos del sistema, 11
- personalización, 12, 24
- tipos de Oracle Solaris, 8
 - diferencias entre ISO y USB, 7
 - ISO para instalación de texto, 8
 - ISO para instalación Live Media, 8
 - ISO para instalaciones automatizadas, 8
 - USB, 7
- imágenes de instalación USB, 7
- imágenes ISO, 8
- instalación automatizada
 - creación de una imagen ISO para, 8
- instalación de prueba
 - creación de una imagen ISO para, 8
- instalación Live Media
 - creación de una imagen ISO para, 8

M

- menú de inicio
 - personalización, 15

P

- paquete de grupo
 - definición de, 18
 - omisión de paquetes en, 18
- paquetes
 - agregación en imagen de instalación, 17
 - desinstalación, 18
 - para instalar, 17
- paquetes SVR4
 - agregación en imagen de instalación, 22
- personalización
 - archivos de manifiesto, 14
 - imágenes de instalación con archivos de manifiesto, 12
 - imágenes de instalación con secuencias de comandos, 24
 - menú de inicio, 15

- puntos de comprobación
 - agregación, 19
 - asignación de nombre, 25
 - campos en, 21
 - definición de, 9
 - secuencias de comandos personalizadas y, 22
 - uso para generar una imagen en etapas, 28
 - uso para hacer referencia a secuencias de comandos personalizadas durante una generación, 24
 - uso para instalar paquetes SVR4, 22

R

- requisitos del sistema para generar imágenes, 11

S

- secuencias de comandos *Ver* secuencias de comandos personalizadas
- secuencias de comandos personalizadas
 - creación y uso, 24
 - puntos de comprobación y, 22