Oracle® VM

Template Builder Installation and User's Guide

Release 2.1

E14391-01

August 2009

Oracle VM Template Builder Installation and User's Guide, Release 2.1

E14391-01

Primary Author: Alison Holloway

Contributor: Tatyana Bagerman, Frank Deng, Chenli Hu, Kevin Lyons.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

4 Managing Resources

5 Administration

6 Creating Templates

7 Creating Template Scripts

8 Deploying Templates

A Command Line Tools

B Template Script Examples

C Troubleshooting

D Oracle VM Template Builder Configuration

Glossary

Index

Preface

The preface contains information on how to use the Oracle VM Template Builder Installation and User's Guide. The areas discussed are:

- Audience
- Documentation Accessibility
- Command Syntax
- Related Documents
- Conventions

Audience

The Oracle VM Template Builder Installation and User's Guide is intended for Oracle VM template builders, and system administrators who want to learn how use to Oracle VM Template Builder to create guest virtual machine templates and deploy in Oracle VM.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at `http://www.fcc.gov/cgb/consumerfacts/trs.html`, and a list of phone numbers is available at `http://www.fcc.gov/cgb/dro/trsphonebk.html`.

Command Syntax

UNIX command syntax appears in `monospace` font. The dollar character ($), number sign (#), or percent character (%) are UNIX command prompts. Do not enter them as part of the command. The following command syntax conventions are used in this Guide:

Convention	Description

backslash \	A backslash is the UNIX command continuation character. It is used in command examples that are too long to fit on a single line. Enter the command as displayed (with a backslash) or enter it on a single line without a backslash:
	`dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \`
	`count=10000`

braces { }	Braces indicate required items: `.DEFINE {macro1}`
brackets [ ]	Brackets indicate optional items: `cvtcrt` *`termname`* `[`*`outfile`*`]`
ellipses ...	Ellipses indicate an arbitrary number of similar items: `CHKVAL fieldname value1 value2 ... valueN`
italics	Italic type indicates a variable. Substitute a value for the variable: *`library_name`*
vertical line |	A vertical line indicates a choice within braces or brackets: `FILE` *`filesize`* `[K|M]`

Related Documents

For more information, see the following documents in the Oracle VM documentation set:

- *Oracle VM Server Quick Start Guide*
- *Oracle VM Server Release Notes*
- *Oracle VM Server Installation Guide*
- *Oracle VM Server User's Guide*
- *Oracle VM Manager Release Notes*
- *Oracle VM Manager Installation Guide*
- *Oracle VM Manager User's Guide*
- *Creating and Using Oracle VM Templates: The Fastest Way to Deploy Any Enterprise Software*, an Oracle white paper:

`http://www.oracle.com/technologies/virtualization/docs/ovmtemplateoverview-whitepaper.pdf`

- *Oracle VM Template Developer's Guide*, an Oracle white paper:

  `http://www.oracle.com/technology/products/vm/pdf/ovmtemplatedevguide-whitepaper.pdf`

You can also get the latest information on Oracle VM by going to the Oracle virtualization Web site:

`http://www.oracle.com/virtualization`

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1 Overview of Oracle VM Template Builder

This Chapter contains an overview of Oracle VM Template Builder. This Chapter contains:

- Overview
- Oracle VM Templates Concepts
- Designing Templates

1.1 Overview

The use of Oracle VM templates for the deployment of applications in Oracle VM guest virtual machines eliminates the need for a user to install and configure the operating system or applications. The templates can simply be downloaded and the guest virtual machines started either from the Oracle VM Manager browser-based interface, or by using an xm command issued at the Oracle VM Server command-line.

Oracle VM Template Builder facilitates the development of Oracle VM templates. It is a kit that includes a web-based application, and a minimum set of the packages for supported guest virtual machine operating systems. Oracle VM Template Builder is itself distributed as a template, and as an RPM.

Installation and configuration of applications and the operating system is eliminated. Creating guest virtual machine templates for your entire stack of applications, operating system and hardware reduces the deployment time for guest virtual machines.

The operating system and applications are configured when the template is started (booted). The configuration can be done based on certain default values and actions, or dynamically based on the user's input. For example, to include a static IP address in a guest virtual machine, you can prompt the user for an IP address, gateway, DNS server, and netmask. These values are then added to the guest virtual machine. The nondefault settings which require user input are created during development of the template.

1.2 Oracle VM Templates Concepts

A template is a snapshot of a preconfigured virtual machine. To create this snapshot Oracle VM Template Builder:

- Creates the virtual machine with the desired operating system
- Installs and configures the target software

- If required, implements the dynamic (initial boot-time) configuration for the operating system and software

The virtual machine with the desired operating system is created from scratch. Oracle VM Template Builder builds the template for you, removing the need to perform these steps manually.

Oracle VM Template Builder uses “Just enough OS” or “JeOS” to facilitate building an operating system instance with only the absolute minimum packages you need for your template. This helps reduce the disk footprint by up to 2GB or more per guest virtual machine. It also improves security and reliability of the guest virtual machine. You can add further packages to customize your guest virtual machine.

Dynamic boot time configuration actions can be implemented in scripts. The images created with Oracle VM Template Builder include an RPM, named ovm-template-config.rpm, which performs operating system configuration. The ovm-template-config RPM can also run your own template scripts to configure the application software installed in the template. See Chapter 7, "Creating Template Scripts" for information on creating template scripts.

A template created with Oracle VM Template Builder contains one guest virtual machine, and consists of one or more binary files, and a text file. The binary files are the disk images of the guest virtual machine. The text file is a virtual machine configuration file (*vm.cfg*). Oracle VM Template Builder creates a single file archive of these files as a *.tar.gz* file. A template created with Oracle VM Template Builder may only contain one guest virtual machine. For example, a template might contain:

```
oracle11g
          /system.img      (disk image with operating system)
          /oracle11g.img   (disk image with Oracle software)
          /vm.cfg          (guest configuration file)
```

Future releases of Oracle VM Template Builder may allow you to create templates with multiple guest virtual machines.

1.3 Designing Templates

A template may support many different use cases, customer requirements, and topologies, or it may be specific to a certain need. The template, when deployed, may be required to support standard application and operating system procedures going forward, for example:

- Patch updates
- Upgrades
- Configuration
- Technical support access

You should consider the requirements of the template before you begin. For example:

- Operating system type, and version
- Paravirtualized or hardware virtualized guest
- Software applications
- Software configuration
- Software installation directory
- Software installation partition size

- Number of virtual CPUs
- Virtual memory size
- Disk swap size
- Root partition size after operating system is installed
- Disk image mount point(s)
- Template name (the default name for the archive and the guest virtual machine)
- Anything specific for your software applications or operating system, for example a database repository

To maximize the performance of guest virtual machines, Oracle recommends you create paravirtualized guest virtual machines in templates.

2 Installing Oracle VM Template Builder

This Chapter contains information on installing, configuring and logging in to Oracle VM Template Builder. This Chapter contains:

2.1 Prerequisites

To install and use Oracle VM Template Builder, you need the following prerequisites, which can be downloaded from `http://www.oracle.com/edelivery`:

- Oracle VM Server
- Oracle VM Manager. This is an optional, but recommended, prerequisite to deploy templates using Oracle VM Manager.
- Oracle VM Template Builder template. The template requires 1GB of RAM and 60GB of disk space on an Oracle VM Server.

2.2 Installing the Oracle VM Template Builder Template

You can install Oracle VM Template Builder from a template, or as an RPM-based install on and computer running Oracle Enterprise Linux. This Guide assumes you are using the template-based install, which is the preferred installation type. This section describes the Oracle VM Template Builder template installation. For information on installing Oracle VM Template Builder as an RPM-based install, see Section 2.10, "Installing Oracle VM Template Builder on Oracle Enterprise Linux as an RPM".

To install the Oracle VM Template Builder template:

1. Copy the downloaded Oracle VM Template Builder template file to the /OVS/seed_pool directory of an Oracle VM Server.
2. Unzip the template file:

   ```
   # cd /OVS/seed_pool
   # unzip OVM_EL5U2_X86_64_TMPLBUILDER_PVM.zip
   ```

   This step creates an OVM_EL5U2_X86_64_TMPLBUILDER_PVM.tgz file.
3. Uncompress the template file:

   ```
   # tar -xzf OVM_EL5U2_X86_64_TMPLBUILDER_PVM.tgz
   ```

   This step creates an OVM_EL5U2_X86_64_TMPLBUILDER_PVM directory under the /OVS/seed_pool directory.

The template is ready to be imported into Oracle VM Manager and used to create the Oracle VM Template Builder guest virtual machine. You can also use the template to create the Oracle VM Template Builder guest virtual machine in Oracle VM Server. Oracle recommends you use Oracle VM Manager to create and manage the Oracle VM Template Builder guest virtual machine.

2.3 Creating the Oracle VM Template Builder Guest Virtual Machine

You can import the template and create the guest virtual machine using Oracle VM Manager. Alternatively, you can use Oracle VM Server to create the guest virtual machine with the xm command-line tool. Oracle recommends you use Oracle VM Manager to import, create and manage guest virtual machines, although this section discusses both options.

2.3.1 Creating the Guest Virtual Machine With Oracle VM Manager

To create the Oracle VM Template Builder guest virtual machine with Oracle VM Manager, you must first import the template into Oracle VM Manager, then create the guest virtual machine. To import the template into Oracle VM Manager and create the virtual machine:

1. Log in to Oracle VM Manager and navigate to the **Resources** tab. The **Virtual Machine Templates** page is displayed. Click **Import**. The **Source** page is displayed.
2. Choose **Select from Server Pool (Discover and register)** and click **Next**. The **General Information** page is displayed.
3. Enter or select the following information:

   Server Pool Name: Select the Server Pool on which the virtual machine is to be located.

   Virtual Machine Template Name: Select the Oracle VM Template Builder template.

   Operating System: Select **Oracle Enterprise Linux 5 64-bit** as the operating system of the virtual machine.

   Virtual Machine System Username: Enter the username **root**, which is used to log in to the virtual machine.

   Virtual Machine System Password: Enter the password **ovsroot**, which is used to log in to the virtual machine.

Description: Enter a description of the virtual machine.

Click **Next**. The **Confirm Information** page is displayed.

4. Click **Confirm**. The **Virtual Machine Template** page is displayed with a message to confirm the template is imported.
5. To make the virtual machine template available for use, select the virtual machine template and click Approve. The **View Virtual Machine Template** page is displayed. Click Approve.
6. Click the **Virtual Machines** tab. The **Virtual Machines** page is displayed.
7. Click **Create Virtual Machine**. The **Creation Method** page is displayed.
8. Select **Create virtual machine based on virtual machine template**. Click **Next**. The **Server Pool** page is displayed.
9. Select the Server Pool in which to create the virtual machine. Select an option from the Preferred Server drop down. Click **Next**. The **Source** page is displayed.
10. Select the template you imported in Step 1, and click **Next**. The **Virtual Machine Information** page is displayed.
11. Enter the virtual machine name in the **Virtual Machine Name** field.

    Enter a password for the console in the **Console Password** field.

    Confirm the console password by entering it again in the **Confirm Console Password** field.

    Select the **Network Interface Card**. Click **Next**. The **Confirm Information** page is displayed.
12. Confirm the virtual machine information, and click **Confirm**. The **Virtual Machine** page is displayed with the message **Creating Virtual Machine**.
13. When the virtual machine is created, the Status changes from **Creating** to **Powered Off**. To power on the virtual machine, click **Power On**.

The Oracle VM Template Builder guest virtual machine is started.

When the Oracle VM Template Builder guest virtual machine is started for the first time, you must configure the guest virtual machine networking. See Section 2.3.3, "Configuring Oracle VM Template Builder" for information on how to configure the networking.

2.3.2 Creating the Guest Virtual Machine With Oracle VM Server

If the Oracle VM Template Builder template is imported and the virtual machine created by Oracle VM Manager, the network MAC address is configured automatically by Oracle VM Manager. If the template is used to create a virtual machine from the Oracle VM Server command line, the network MAC address must first be configured before the guest virtual machine is started.

To configure the network MAC address of the guest virtual machine in Oracle VM Server:

1. Change directory to where the virtual machine files are located:

```
# cd /OVS/seed_pool/OVM_EL5U2_X86_64_TMPLBUILDER_PVM
```

2. Run the following command to generate a new MAC address:

```
# PYTHONPATH=/opt/ovs-agent-2.2 python -c "from OVSCommons import randomMAC;
print randomMAC()"
```

3. Edit the */OVS/seed_pool/OVM_EL5U2_X86_64_TMPLBUILDER_PVM/vm.cfg* file and change the line starting with `'vif'` to:

   ```
   vif = [ 'mac=00:16:3E:xx:xx:xx', ]
   ```

   Where `00:16:3E:xx:xx:xx` is the MAC address generated in step 2.

The network MAC address is now configured for the guest virtual machine.

To create and start the guest virtual machine, enter:

```
# xm create vm.cfg
```

The Oracle VM Template Builder guest virtual machine is started.

When Oracle VM Template Builder is started for the first time, you must configure the guest virtual machine networking. See Section 2.3.3, "Configuring Oracle VM Template Builder" for information on how to configure the networking.

2.3.3 Configuring Oracle VM Template Builder

When Oracle VM Template Builder is started for the first time, you must configure the guest virtual machine. Prompts are displayed for the configuration on the initial startup of the guest virtual machine. To configure Oracle VM Template Builder:

1. Log in to the Oracle VM Template Builder guest virtual machine console. If you are using Oracle VM Manager, click the **Console** button on the **Virtual Machines** screen. If you are using Oracle VM Server, connect to the VNC session using VNCViewer. See the *Oracle VM Server User's Guide* for more information on connecting to a guest virtual machine using VNC.

2. The following question is displayed requesting whether to use a static IP address, or have DHCP assign one:

   ```
   Do you want to enable dynamic IP configuration (DHCP) Y/n?
   ```

   The decision to use DHCP or a static IP address depends entirely on the requirements of your template. Enter `Y` to use DHCP, or `n` to use a static IP address. If you do not enter anything, after a period of time, DHCP is selected automatically.

3. If you selected to use DHCP to assign an IP address, an IP address is acquired and configured in the guest. If you selected to use a static IP address, the following questions are displayed:

   ```
   Enter a static IP addresss:
   Enter netmask:
   Enter gateway:
   Enter DNS server:
   Enter hostname (e.g, host.domain.com):
   ```

   Enter a static IP address, the netmask for your network, the IP address for the default gateway for your network, the IP address for the DNS server for your network, and the host name to use for the guest virtual machine.

4. The Oracle VM Template Builder configuration completes, and displays a message with the URL to use to connect to Oracle VM Template Builder web interface. Press any key to acknowledge the message. The Oracle VM Template Builder command-line login screen is displayed.

Oracle VM Template Builder network configuration is complete. You can now log in to the Oracle VM Template Builder web interface.

2.4 Accessing Oracle VM Template Builder Web Interface

To access Oracle VM Template Builder web interface, open a web browser and enter the URL for the application:

```
http://host/ovmtb
```

The Oracle VM Template Builder login screen is displayed.

2.5 Creating an Oracle VM Template Builder User

There is no default Oracle VM Template Builder user. You must create one or more using Oracle VM Template Builder. There are severalseveral methods to create a user:

- Using the Login screen. See Section 3.2, "Registering a User".
- Using the Administration screen. See Section 5.1.1, "Creating a User".

2.6 Logging in to Oracle VM Template Builder Web Interface

To log in to Oracle VM Template Builder, enter your username and password in the **Login** screen and click **Login**. You are logged into Oracle VM Template Builder. See Section 3.1, "Logging In" in Chapter 3, "Logging In" for more information on logging in.

If you do not have a username and password for Oracle VM Template Builder, see Section 2.5, "Creating an Oracle VM Template Builder User".

2.7 Logging in to Oracle VM Template Builder Guest Virtual Machine

To log in to the Oracle VM Template Builder guest virtual machine operating system:

1. Log in to Oracle VM Manager.
2. Select the Oracle VM Template Builder virtual machine in the Virtual Machines list. Click **Console**. The guest virtual machine console is displayed.
3. Enter the console password. The console password is the password you set when you created the guest virtual machine. The guest login screen is displayed.
4. Log in to the Oracle VM Template Builder guest virtual machine.

> **Note:** The default operating system username and password for the Oracle VM Template Builder guest virtual machine is:
>
> Username: root
>
> Password: ovsroot

You are logged into the Oracle VM Template Builder guest virtual machine operating system.

2.8 Starting and Stopping Oracle VM Template Builder

To start and stop Oracle VM Template Builder, log in to the Oracle VM Template Builder guest virtual machine and run the appropriate start or stop command. To log in to the guest virtual machine, see Section 2.7, "Logging in to Oracle VM Template Builder Guest Virtual Machine".

The Oracle VM Template Builder service (ovmtb) starts the daemon which performs the template builds. The ovmtb service is separate to the web server (httpd) service, which serves up the Oracle VM Template Builder user interface. You must have both the ovmtb and httpd services running to use Oracle VM Template Builder.

To start the Oracle VM Template Builder service, enter:

```
# service ovmtb start
```

To start the web server service, enter:

```
# service httpd start
```

To stop the Oracle VM Template Builder service, enter:

```
# service ovmtb stop
```

To stop the web server service, enter:

```
# service httpd stop
```

To restart Oracle VM Template Builder service, enter:

```
# service ovmtb restart
```

To restart the web server service, enter:

```
# service httpd restart
```

2.9 Customizing the Oracle VM Template Builder Guest Virtual Machine

You can change some configuration parameters for the Oracle VM Template Builder guest virtual machine, such as the number of VCPUs, RAM, and whether to enable High Availability. See the *Oracle VM Manager User's Guide* for more information. Note that changing these settings may affect the performance of Oracle VM Template Builder.

You can also increase the disk size for the Oracle VM Template Builder guest virtual machine. You may be required increase the disk size if you create many Oracle VM templates and the default disk size is not sufficient for your needs. This section discusses adding a new disk to the Oracle VM Template Builder guest virtual machine using a virtual disk or an NFS share.

These steps assume you are using the Oracle VM Template Builder template, not the RPM-based installation. If you performed an RPM installation the steps may differ, depending on your system configuration.

2.9.1 Adding a Virtual Disk

To add a new virtual disk, on the Oracle VM Template Builder guest virtual machine:

1. Shut down the Oracle VM Template Builder guest virtual machine.
2. Add a new virtual disk to the Oracle VM Template Builder guest virtual machine using Oracle VM Manager.

3. Start the Oracle VM Template Builder guest virtual machine.

4. Log in to the Oracle VM Template Builder guest virtual machine. Create a partition on the new virtual disk using the fdisk utility:

   ```
   # fdisk /dev/xvdc
   ```

   The following prompt is displayed:

   ```
   > Command (m for help): n
   ```

   Enter `n` for new partition and press **Enter**. The following prompt is displayed:

   ```
   > Command action
   >    e   extended
   >    p   primary partition (1-4)
   > p
   ```

   Enter `p` for a primary partition and press **Enter**. The following prompt is displayed:

   ```
   > Partition number (1-4):
   > 1
   ```

   Enter `1` and press **Enter**.

   Press **Enter** twice to accept the default starting and ending locations and to create a partition that spans the entire disk. The following prompt is displayed:

   ```
   > Command (m for help): w
   ```

   Enter `w` to and press **Enter** to write the partition table to the disk and to exit fdisk utility.

5. Format the new partition using the mkfs utility:

   ```
   # mkfs -t ext3 /dev/xvdc1
   ```

6. Stop the Oracle VM Template Builder service, and the web server service:

   ```
   # service httpd stop
   # service ovmtb stop
   ```

7. Remount the data on the virtual disk image:

   ```
   # umount /opt/ovmtb
   # mkdir /opt/ovmtb.bak
   # mount /dev/xvdb1 /opt/ovmtb.bak
   ```

8. Mount the new data partition on /opt/ovmtb, for example:

   ```
   # mount /dev/xvdc1 /opt/ovmtb
   ```

9. Copy the contents of the old data partition to /opt/ovmtb:

   ```
   # cp -a /opt/ovmtb.bak/* /opt/ovmtb
   ```

10. Restart the Oracle VM Template Builder service, and the web server service:

    ```
    # service httpd start
    # service ovmtb start
    ```

11. Edit the /etc/fstab file so that the virtual disk (and not /dev/xvdb1) is mounted on /opt/ovmtb when the system restarts, for example:

```
/dev/xvdc1        /opt/ovmtb     ext3      defaults        1    2
```

The new disk is available in the guest virtual machine, and mounted as /opt/ovmtb, which is the location where Oracle VM Template Builder builds and saves templates.

2.9.2 Adding an NFS Share Disk

To add a new disk using NFS share, on the Oracle VM Template Builder guest virtual machine:

1. Stop the Oracle VM Template Builder service, and the web server service:

   ```
   # service ovmtb stop
   # service httpd stop
   ```

2. Remount the data virtual disk image:

   ```
   # umount /opt/ovmtb
   # mkdir /opt/ovmtb.bak
   # mount /dev/xvdb1 /opt/ovmtb.bak
   ```

   The application data is now located at /opt/ovmtb.bak.

3. Mount the NFS share on /opt/ovmtb. For example, to mount an NFS volume shared as /vol from server nfs-server:

   ```
   # mount nfs-server:/vol /opt/ovmtb
   ```

4. Copy the contents of /opt/ovmtb.bak to /opt/ovmtb:

   ```
   # cp -a /opt/ovmtb.bak/* /opt/ovmtb
   ```

5. Start the Oracle VM Template Builder service, and the web server service:

   ```
   # service httpd start
   # service ovmtb start
   ```

6. Edit the /etc/fstab file so that the NFS share (and not /dev/xvdb1) is mounted on /opt/ovmtb when the system restarts.

The new disk is available in the guest virtual machine, and mounted as /opt/ovmtb, which is the location where Oracle VM Template Builder builds and saves templates.

2.10 Installing Oracle VM Template Builder on Oracle Enterprise Linux as an RPM

You can install Oracle VM Template Builder as an RPM-based install on a computer running Oracle Enterprise Linux, in addition to as a template. You need Oracle Enterprise Linux Release 5 Update 2 (or later) to perform an RPM-based install. If you install on a 32-bit operating system, you can only create 32-bit templates. If you install on a 64-bit operating system, you can create both 32- and 64-bit templates.

This Guide assumes you are using the Oracle VM Template Builder template, which is the preferred installation type. If you choose to perform an RPM-based installation, some information in this Guide may differ to your environment.

To install Oracle VM Template Builder as an RPM-based install:

1. Make sure there is enough space in the /opt directory to hold the templates you intend to create.

2. Download and install any prerequisite Oracle Enterprise Linux packages not installed on your computer. The packages you need are:

   - bzip2-1.0.3-3
   - createrepo-0.4.11-3
   - httpd-2.2.3-11.el5_1.3.0.1
   - nfs-utils-1.0.9-33
   - mod_python-3.2.8-3.1
   - parted-1.8.1-17
   - perl-5.8.8-10
   - python-2.4.3-21
   - sendmail-8.13.8-2
   - sqlite-3.3.6-2
   - unzip-5.52-2.2.1
   - wget-1.10.2-7
   - yum-utils-1.1.10-9

   You can install these packages from the public Yum repository at:

   http://public-yum.oracle.com

   If you have configured your operating system to use a Yum repository, you can install these packages with the following command:

   ```
   # yum install bzip2 createrepo httpd nfs-utils mod_python parted perl python
   sendmail sqlite unzip wget yum-utils
   ```

   If you downloaded these packages, you can install them with the following command for each RPM:

   ```
   # rpm -ivh name.rpm
   ```

   For example

   ```
   # rpm -ivh yum-utils-1.1.16-13.el5.noarch.rpm
   ```

3. Download JeOS and the JeOS templates from:

   http://edelivery.oracle.com/linux

   Select one or both of the following from the eDelivery Web site, according to your template build requirements:

   - **Oracle Enterprise Linux** > **Oracle Enterprise Linux JeOS for Building Oracle VM templates Media Pack v2 for x86 (32 bit)**
   - **Oracle Enterprise Linux** > **Oracle Enterprise Linux JeOS for Building Oracle VM templates Media Pack v2 for x86_64 (64 bit)**

   You should download the JeOS base image and JeOS templates for each operating system platform on which you want to create Oracle VM Template Builder templates. Under each location above, download:

   - The *JeOS base* image
   - All the JeOS template images you require for your template builds

4. Unzip the downloaded files to extract the RPMs.
5. Install JeOS:

   ```
   # rpm -ivh ovm-modify-jeos-version.rpm
   ```

   The ovm-template-config-version.noarch.rpm file is installed during the installation of ovm-modify-jeos-version.noarch.rpm and does not need to be installed manually.
6. Install any JeOS templates you downloaded:

   ```
   # rpm -ivh ovm-name.rpm
   ```

   For example

   ```
   # rpm -ivh ovm-el5u2-xvm-jeos-1.0.1-6.el5.x86_64.rpm
   ```

7. Download the prerequisite third party packages:
   - python-simplejson
   - python-sqlite2
   - Django
8. Install each third party package with the command:

   ```
   # rpm -ivh name.rpm
   ```

   For example

   ```
   # rpm -ivh python-sqlite2-2.3.3-1.el5.x86_64.rpm
   ```

9. Download and install Oracle VM Template Builder from any of these locations:

   http://edelivery.oracle.com

   http://public-yum.oracle.com

   http://oss.oracle.com

   If you have configured your operating system to use a Yum repository, you can install Oracle VM Template Builder with the following commands:

   ```
   # yum install ovmtb
   # yum install ovmtb-images
   ```

   If you downloaded the packages, you can install them with the following commands:

   ```
   # rpm -ivh ovmtb-version.rpm
   # rpm -ivh ovmtb-images-version.rpm
   ```

10. Restart the Oracle VM Template Builder service and the web server service:

    ```
    # service httpd restart
    # service ovmtb restart
    ```

11. Make sure the http and ssh ports are open to access the web interface and command shell with the command:

    ```
    # system-config-securitylevel-tui --quiet --port=http --port=ssh
    ```

Oracle VM Template Builder is installed and started.

If you cannot access the web interface, you may want to run the Oracle VM Template Builder configuration script. As the *root* user, enter

```
# /var/www/html/ovmtb/scripts/ovmtb-reconfig.sh
```

2.11 Upgrading JeOS and JeOS Templates

You can upgrade JeOS and JeOS templates independently to upgrading Oracle VM Template Builder. Oracle VM Template Builder automatically uses any updated JeOS RPM and JeOS templates. JeOS is the tool which creates the Oracle VM Template Builder templates, based on the available JeOS templates. You can upgrade JeOS independently to the JeOS templates, or you can upgrade the JeOS templates independently to JeOS.

This upgrade procedure assumes you are upgrading both JeOS and the JeOS templates. To upgrade JeOS and JeOS templates:

1. Download JeOS and the JeOS templates from:

   http://edelivery.oracle.com/linux

   Select one or both of the following from the eDelivery Web site, according to your needs:

   - **Oracle Enterprise Linux** > **Oracle Enterprise Linux JeOS for Building Oracle VM templates Media Pack v2 for x86 (32 bit)**
   - **Oracle Enterprise Linux** > **Oracle Enterprise Linux JeOS for Building Oracle VM templates Media Pack v2 for x86_64 (64 bit)**

   For each operating system platform that you want to upgrade:

   - Download the *JeOS base* image to upgrade JeOS
   - Download all the JeOS template images you require for your template builds.

2. Unzip the downloaded files to extract the RPMs.
3. If you are upgrading JeOS, run the upgrade command as the *root* user:

   ```
   # rpm -U ovm-modify-jeos_version.noarch.rpm
   ```

4. If you are installing any new JeOS templates, run the following command for each new template:

   ```
   # rpm -ivh ovm-template_name.rpm
   ```

   For example

   ```
   # rpm -ivh ovm-el5u2-xvm-jeos-1.0.1-6.el5.x86_64.rpm
   ```

5. If you are upgrading any JeOS templates, run the following command for each template you are upgrading:

   ```
   # rpm -U ovm-template_name.rpm
   ```

   For example

   ```
   # rpm -U ovm-el5u2-xvm-jeos-1.1.0-3.el5.x86_64.rpm
   ```

JeOS is upgraded and any new or upgraded JeOS templates are installed. Oracle VM Template Builder automatically picks up the JeOS update and any new or updated JeOS templates.

3 Logging In

This Chapter describes registering a user and logging in to the web interface. This Chapter contains:

- Logging In
- Registering a User
- Resetting a Password

3.1 Logging In

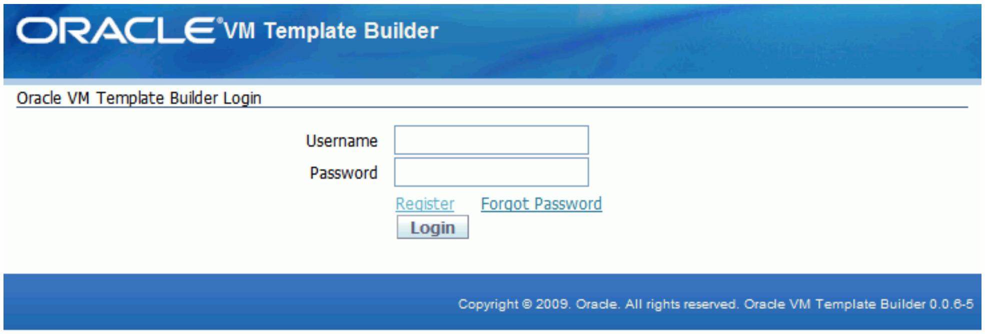
To log in to Oracle VM Template Builder:

1. Open a web browser and enter the URL for the application:

   `http://host/ovmtb`

   The Oracle VM Template Builder **Login** screen is displayed.

Figure 3–1 Login Screen


ORACLE VM Template Builder
Oracle VM Template Builder Login
Username
Password
Register
Forgot Password
Login
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

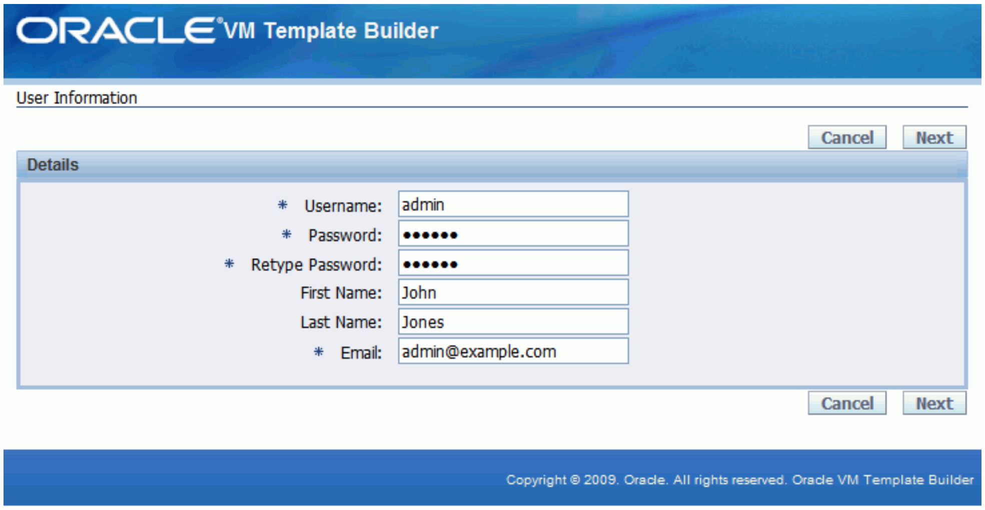
2. Enter your username in the **Username** field, and your password in the **Password** field, and click **Login**. You are logged into Oracle VM Template Builder. If you do not have a login, see Registering a User.

3.2 Registering a User

There is no default user for Oracle VM Template Builder. Before you can log in, you must register a user. To create a user from the **Login** screen:

1. Click the **Register** link on the **Login** screen. The **User Information** screen is displayed.

Figure 3–2 User Information Screen



* Username: admin

* Password: ••••••

* Retype Password: ••••••

First Name: John

Last Name: Jones

* Email: admin@example.com

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder

2. Enter the following information:

   - **Username**: A name for the user.
   - **Password**: A password for the user.
   - **Retype Password**: Confirm the password by entering it again.
   - **First Name**: The first name of the user.
   - **Last Name**: The last name of the user.
   - **Email**: An email address for the user. This email address is used when resetting the user's password.

   Click **Next**. The **Confirm Information** screen is displayed.

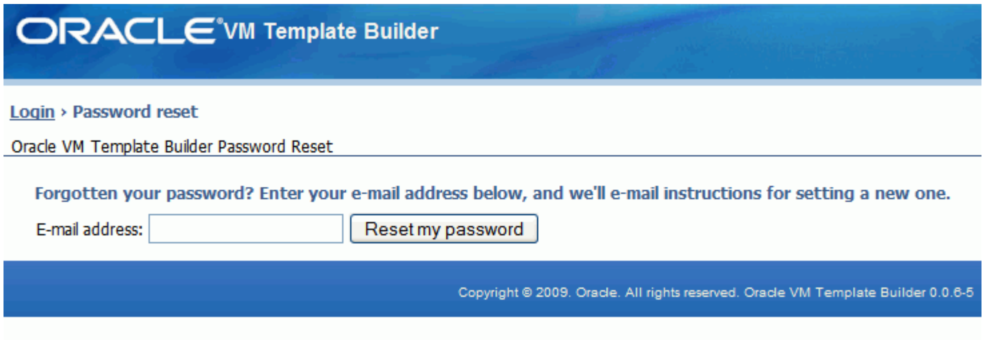
3. The **Cancel** button cancels the user creation. The **Previous** button returns you to the previous screen. Click **Confirm** to confirm the user information. The user is created and the Login screen is displayed.

3.3 Resetting a Password

To reset your login password from the **Login** screen:

1. Click the **Forgot Password** link. The **Password Reset** screen is displayed.

Figure 3–3 Password Reset Screen



Oracle VM Template Builder Password Reset

Forgotten your password? Enter your e-mail address below, and we'll e-mail instructions for setting a new one.

E-mail address:

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

2. Enter the email address used to create the user in the **Email address** field. Click **Reset my password**. An email is set to your email address which provides instructions on resetting your password.

4

Managing Resources

This Chapter describes the resources used in Oracle VM Template Builder, such as ISO files, Yum repositories and JeOS images. It discusses creating and editing these resources. At least one ISO file, or one Yum repository must be configured before creating a template. This Chapter contains:

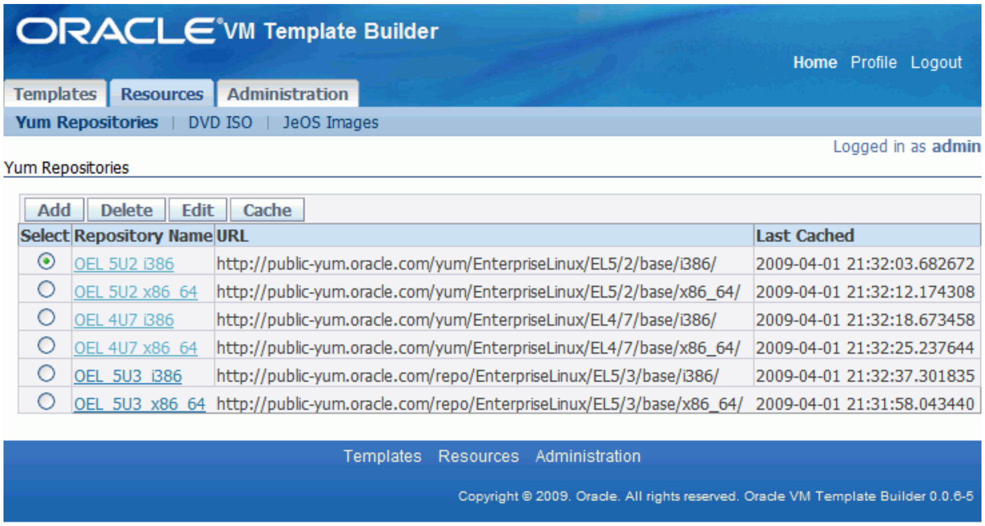
- Yum Repositories
- DVD ISO Files
- JeOS Images

Oracle VM Template Builder resources are created and managed using the **Resources** tab. To perform the actions discussed in this Chapter, select the **Resources** tab.

4.1 Yum Repositories

When creating templates, a Yum repository can be used to resolve and download any RPM dependencies that arise during the installation of the operating system of the guest virtual machine. You can manage Yum repositories with the **Yum Repositories** sub-tab in the **Resources** tab.

Figure 4–1 Yum Repositories Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Yum Repositories | DVD ISO | JeOS Images
Logged in as admin
Add Delete Edit Cache
Templates Resources Administration

Yum Repositories

Select	Repository Name	URL	Last Cached
●	OEL 5U2 i386	http://public-yum.oracle.com/yum/EnterpriseLinux/EL5/2/base/i386/	2009-04-01 21:32:03.682672
○	OEL 5U2 x86 64	http://public-yum.oracle.com/yum/EnterpriseLinux/EL5/2/base/x86_64/	2009-04-01 21:32:12.174308
○	OEL 4U7 i386	http://public-yum.oracle.com/yum/EnterpriseLinux/EL4/7/base/i386/	2009-04-01 21:32:18.673458
○	OEL 4U7 x86 64	http://public-yum.oracle.com/yum/EnterpriseLinux/EL4/7/base/x86_64/	2009-04-01 21:32:25.237644
○	OEL 5U3 i386	http://public-yum.oracle.com/repo/EnterpriseLinux/EL5/3/base/i386/	2009-04-01 21:32:37.301835
○	OEL 5U3 x86 64	http://public-yum.oracle.com/repo/EnterpriseLinux/EL5/3/base/x86_64/	2009-04-01 21:31:58.043440

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

The Yum Repositories screen displays a table with all of the Yum repositories configured in the system. The table displays columns listing the repository name, URL, and the date the repository was last cached. At the upper left hand corner of the table are buttons that can be used to add, delete, edit, and cache the repositories.

There are several default Yum repositories included for Oracle Enterprise Linux releases and architectures. These repositories include the full releases of Oracle Enterprise Linux and mirror those releases on the Oracle Enterprise Linux DVDs. The repositories are available at Oracle's public Yum repository:

http://public-yum.oracle.com

No patches are included in these repositories. To access patches for these releases, you should contact Oracle to purchase an Oracle Enterprise Linux Support contract and gain access to the repository which contains these patches. The default Yum repositories included in this release of Oracle VM Template Builder are:

- Oracle Enterprise Linux Release 4 Update 7 i386
- Oracle Enterprise Linux Release 4 Update 7 x86_64
- Oracle Enterprise Linux Release 5 Update 2 i386
- Oracle Enterprise Linux Release 5 Update 2 x86_64

To create your own repository, see the article *Yum Repository Setup* on the Oracle Technology Network:

http://www.oracle.com/technology/tech/linux/htdocs/yum-repository-setup.html

To add a Yum repository that is located outside your firewall, you must add your proxy server to the */etc/yum.conf* file. For example, a proxy server entry in the *yum.conf* file might contain:

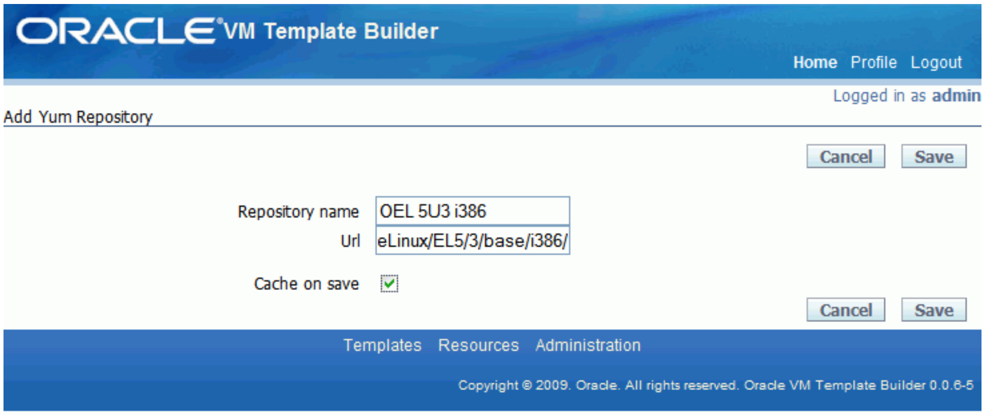
```
proxy = myproxy.example.com
proxy_username = myusername
proxy_password = mypassword
```

4.1.1 Adding a Yum Repository

To add a Yum repository:

1. Click **Add** on the **Yum Repositories** screen. The **Add Yum Repository** screen is displayed.

Figure 4–2 Add Yum Repository Screen


ORACLE VM Template Builder
Home
Profile
Logout
Logged in as admin
Add Yum Repository
Cancel
Save
Cancel
Save
Templates
Resources
Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

Repository name OEL 5U3 i386
Url eLinux/EL5/3/base/i386/
Cache on save ☑

2. Enter or select the following information:

- **Repository name**: A name for the repository.
- **URL**: The URL for the repository. You can enter an HTTP or FTP location for the repository. NFS and file repositories are not supported in this release.
- **Cache on save**: Whether to cache the repository when it is saved. Selecting this option creates a cache of the RPMs in the repository, removing the need to read the repository contents each time a template is created.

Click **Save** to add the repository. The **Yum Repositories** screen is displayed listing the new repository.

4.1.2 Editing a Yum Repository

To edit a Yum repository, select the repository in the list and click **Edit**, or click the **Repository Name** in the table. The **Edit Yum Repository** screen is displayed. Edit the information for the repository.

Click **Save** to save the changes. The **Yum Repositories** screen is displayed.

4.1.3 Deleting a Yum Repository

To delete a Yum repository, select the repository in the list and click **Delete**. The repository is deleted.

4.1.4 Caching a Yum Repository

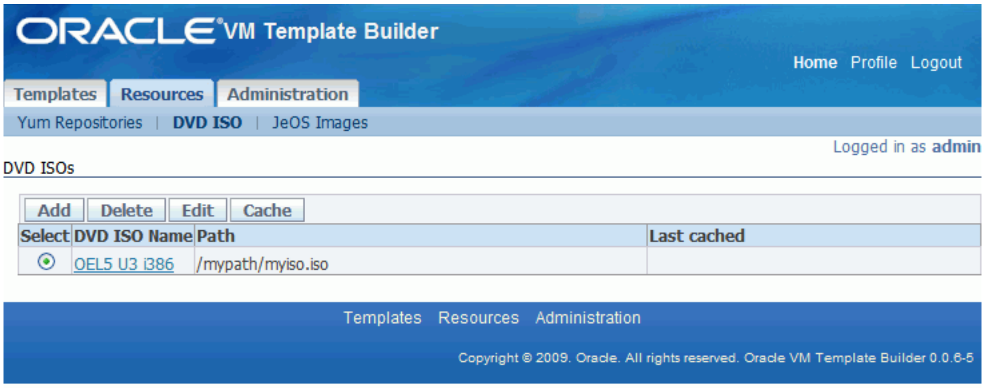
To cache, or refresh the cache of a Yum repository, select the repository in the list and click **Cache**. The DVD ISO cache is created or updated.

4.2 DVD ISO Files

When creating templates, Oracle VM Template Builder can use the extracted contents of an Oracle Enterprise Linux ISO file to resolve any RPM dependencies that arise during the creation of the operating system in the guest virtual machine.

You can manage the DVD (and CD) ISO locations with the **DVD ISO** sub-tab of the **Resources** tab. The **DVD ISOs** screen displays a table listing the DVD ISOs configured in the system.

Figure 4–3 DVD ISOs Screen



Logged in as **admin**

DVD ISOs

Select	DVD ISO Name	Path	Last cached
●	OEL5 U3 i386	/mypath/myiso.iso	

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

The table displays columns listing the ISO name, path, and the date the ISO was last cached. At the upper left hand corner of the table are buttons that can be used to add, delete, edit, and cache the DVD ISOs.

4.2.1 Staging a DVD ISO

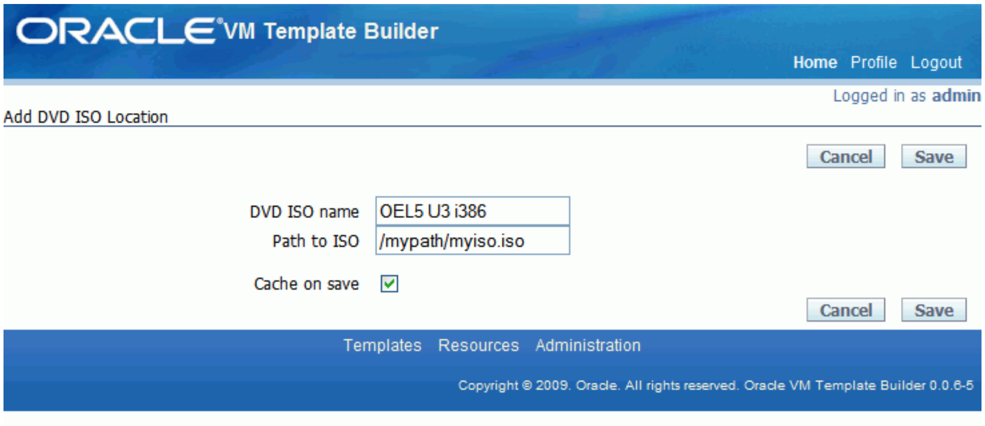
A DVD ISO location consists of an Oracle Enterprise Linux DVD or CD ISO set that has been extracted into a single directory or mounted. To create a DVD ISO location, copy all of the contents of a Oracle Enterprise Linux DVD or CD set into a directory on the guest virtual machine that is running Oracle VM Template Builder, or mount the ISO.

4.2.2 Adding a DVD ISO

To add a DVD ISO location:

1. Click **Add** on the **DVD ISO** screen. The **Add DVD ISO Location** screen is displayed.

Figure 4–4 Add DVD ISO Location Screen



Add DVD ISO Location

DVD ISO name: OEL5 U3 i386

Path to ISO: /mypath/myiso.iso

Cache on save ☑

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

2. Enter or select the following information:

   - **DVD ISO name**: A description of the DVD ISO
   - **Path to ISO**: The path to the extracted DVD on the guest virtual machine on which Oracle VM Template Builder is running.
   - **Cache on save**: Whether to cache the repository when it is saved. Selecting this option creates a cache of the RPMs in the repository, removing the need to read the repository contents each time a template is created.

   Click **Save** to add the DVD ISO. The **DVD ISO** screen is displayed listing the new DVD ISO.

4.2.3 Editing a DVD ISO

To edit a DVD ISO, select the DVD ISO in the list and click **Edit**, or click the **DVD ISO Name** in the table. The **Edit DVD ISO** screen is displayed. Edit the information for the ISO.

Click **Save** to save the changes. The **DVD ISO** screen is displayed.

4.2.4 Deleting a DVD ISO

To delete a DVD ISO, select the DVD ISO in the list and click **Delete**. The ISO is deleted.

4.2.5 Caching a DVD ISO

To cache, or refresh the cache of a DVD ISO, select the DVD ISO in the list and click **Cache**. The DVD ISO cache is created or updated.

4.3 JeOS Images

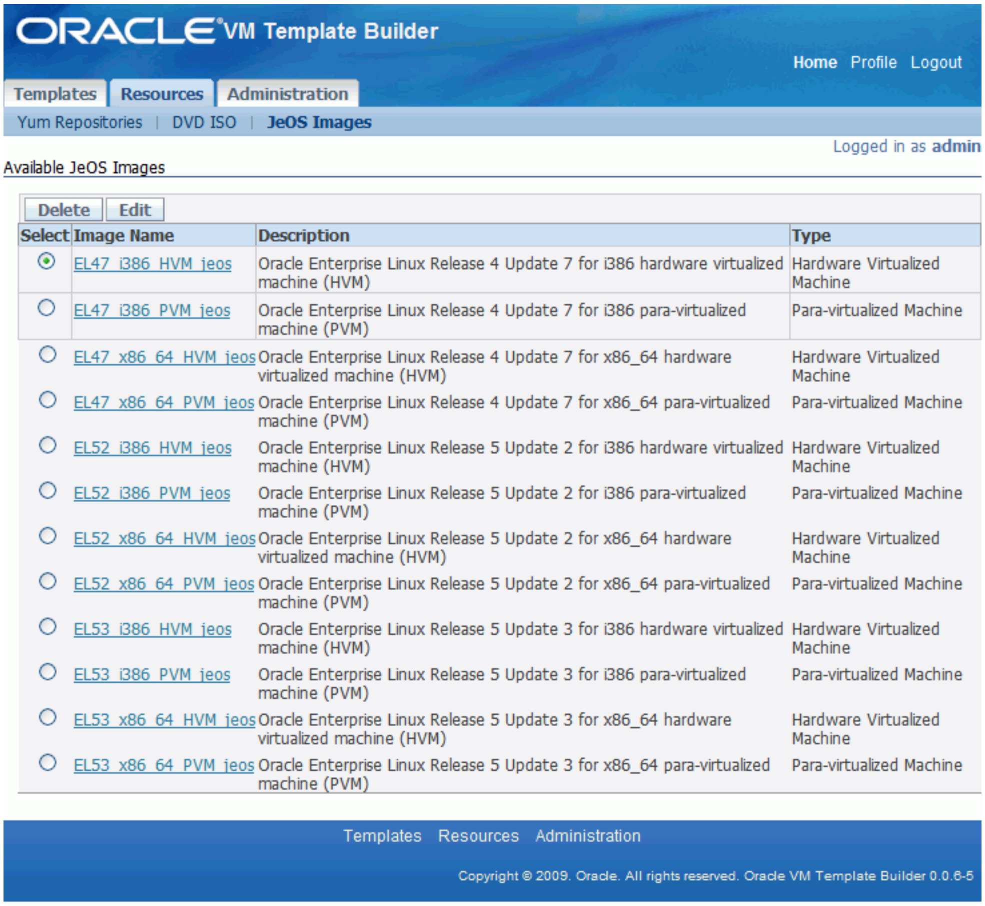
Oracle VM Template Builder uses JeOS to facilitate building an operating system instance with only the absolute minimum packages you need for a template. The operating systems you can select in Oracle VM Template Builder are derived from the available JeOS images. New Oracle VM Template Builder releases and patch updates may include updated JeOS images. You can also upgrade JeOS and update the JeOS images separately to Oracle VM Template Builder. To upgrade JeOS and JeOS images, see Section 2.11, "Upgrading JeOS and JeOS Templates".

The available JeOS images in this release of Oracle VM Template Builder include both hardware virtualized and paravirtualized images for the following operating systems and architectures:

- Oracle Enterprise Linux Release 4 Update 7 i386
- Oracle Enterprise Linux Release 4 Update 7 x86_64
- Oracle Enterprise Linux Release 5 Update 2 i386
- Oracle Enterprise Linux Release 5 Update 2 x86_64
- Oracle Enterprise Linux Release 5 Update 3 i386
- Oracle Enterprise Linux Release 5 Update 3 x86_64

You can manage the JeOS images with the **JeOS Images** sub-tab of the **Resources** tab. The **JeOS Images** screen displays a table listing the JeOS images configured in the system.

Figure 4–5 Available JeOS Images Screen



Available JeOS Images

Select	Image Name	Description	Type
●	EL47_i386_HVM_jeos	Oracle Enterprise Linux Release 4 Update 7 for i386 hardware virtualized machine (HVM)	Hardware Virtualized Machine
○	EL47_i386_PVM_jeos	Oracle Enterprise Linux Release 4 Update 7 for i386 para-virtualized machine (PVM)	Para-virtualized Machine
○	EL47_x86_64_HVM_jeos	Oracle Enterprise Linux Release 4 Update 7 for x86_64 hardware virtualized machine (HVM)	Hardware Virtualized Machine
○	EL47_x86_64_PVM_jeos	Oracle Enterprise Linux Release 4 Update 7 for x86_64 para-virtualized machine (PVM)	Para-virtualized Machine
○	EL52_i386_HVM_jeos	Oracle Enterprise Linux Release 5 Update 2 for i386 hardware virtualized machine (HVM)	Hardware Virtualized Machine
○	EL52_i386_PVM_jeos	Oracle Enterprise Linux Release 5 Update 2 for i386 para-virtualized machine (PVM)	Para-virtualized Machine
○	EL52_x86_64_HVM_jeos	Oracle Enterprise Linux Release 5 Update 2 for x86_64 hardware virtualized machine (HVM)	Hardware Virtualized Machine
○	EL52_x86_64_PVM_jeos	Oracle Enterprise Linux Release 5 Update 2 for x86_64 para-virtualized machine (PVM)	Para-virtualized Machine
○	EL53_i386_HVM_jeos	Oracle Enterprise Linux Release 5 Update 3 for i386 hardware virtualized machine (HVM)	Hardware Virtualized Machine
○	EL53_i386_PVM_jeos	Oracle Enterprise Linux Release 5 Update 3 for i386 para-virtualized machine (PVM)	Para-virtualized Machine
○	EL53_x86_64_HVM_jeos	Oracle Enterprise Linux Release 5 Update 3 for x86_64 hardware virtualized machine (HVM)	Hardware Virtualized Machine
○	EL53_x86_64_PVM_jeos	Oracle Enterprise Linux Release 5 Update 3 for x86_64 para-virtualized machine (PVM)	Para-virtualized Machine

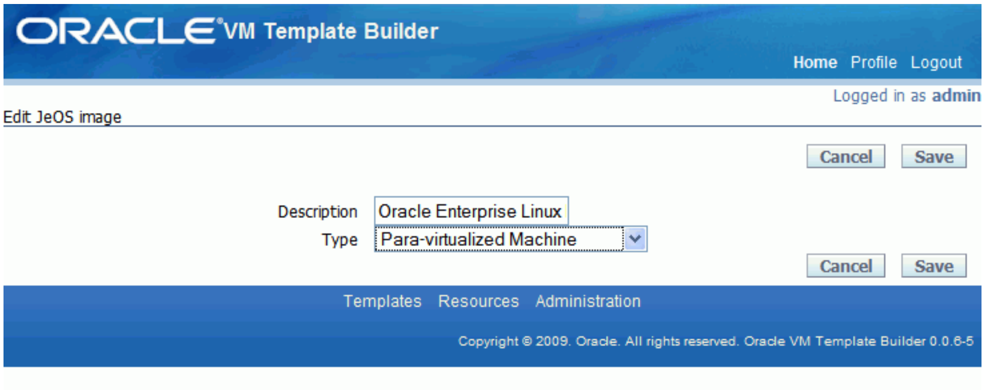
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

The table displays columns listing the image name, description, and type (hardware virtualized or paravirtualized). At the upper left hand corner of the table are buttons that can be used to delete and edit the JeOS images.

4.3.1 Editing a JeOS Image

You can edit a JeOS image to change the description, or whether to create a hardware virtualized or paravirtualized guest virtualized machine. To edit a JeOS image, select the JeOS image in the list and click **Edit**, or click the **Image Name** in the table. The **Edit JeOS Image** screen is displayed.

Figure 4–6 Edit JeOS Image Screen



Edit JeOS image

Description: Oracle Enterprise Linux

Type: Para-virtualized Machine

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

You can change the image description in the **Description** field, and whether the image creates paravirtualized or hardware virtualized guest virtual machines using the **Type** drop down.

Click **Save** to save the changes. The **JeOS Images** screen is displayed.

4.3.2 Deleting a JeOS Image

To delete a JeOS image, select the JeOS image in the list and click **Delete**. The image is deleted.

5 Administration

This Chapter describes the Administration tab. The Administration tab is used to create and manage users and to view the status of the Oracle VM Template Builder daemon. This Chapter contains:

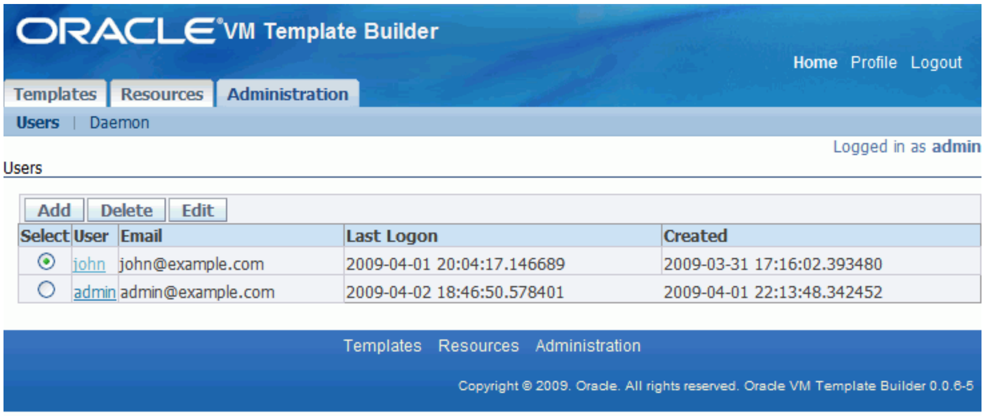
- Managing Users
- Oracle VM Template Builder Daemon

To perform the actions discussed in this Chapter, select the **Administration** tab. The Administration tab contains links to the **Users** screen, and to the **Daemon** screen.

5.1 Managing Users

The **Users** sub-tab displays the **Users** screen and enables you to manage Oracle VM Template Builder users.

Figure 5–1 Users Screen



Select	User	Email	Last Logon	Created
●	john	john@example.com	2009-04-01 20:04:17.146689	2009-03-31 17:16:02.393480
○	admin	admin@example.com	2009-04-02 18:46:50.578401	2009-04-01 22:13:48.342452

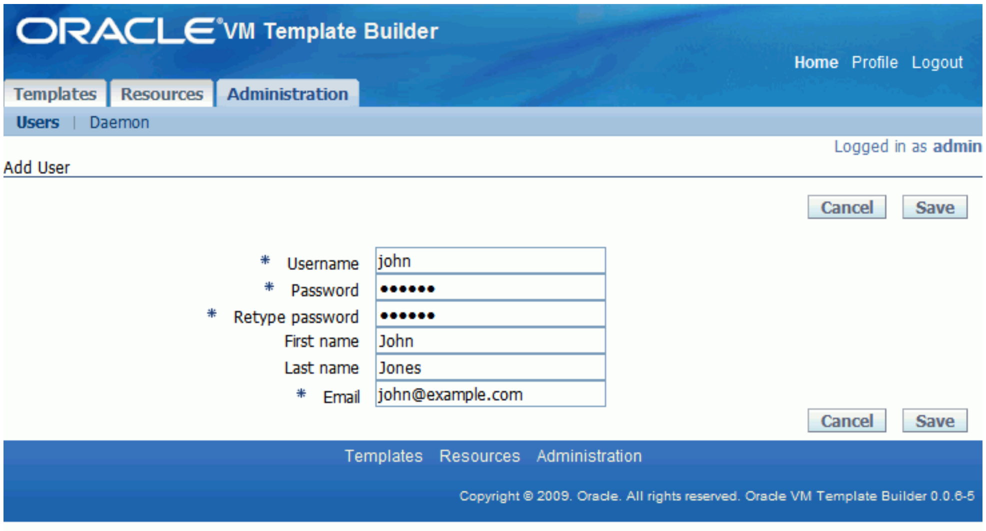
At the top of the Users table is a set of buttons to add, delete and edit Oracle VM Template Builder users. The table displays the existing users, the last time the user logged in, and when the user was created.

5.1.1 Creating a User

To create a user:

1. Click **Add**. The **Add User** screen is displayed.

Figure 5–2 Add User Screen



* Username	john
* Password	••••••
* Retype password	••••••
First name	John
Last name	Jones
* Email	john@example.com

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

2. Enter or select the following information:

   - **Username**: A name for the user.
   - **Password**: A password for the user.
   - **Retype Password**: Confirm the password by entering it again.
   - **First Name**: The first name of the user.
   - **Last Name**: The last name of the user.
   - **Email**: An email address for the user. This email address is used when resetting the user's password.

   Click **Save**. The **Users** screen is displayed and lists the new user.

5.1.2 Deleting a User

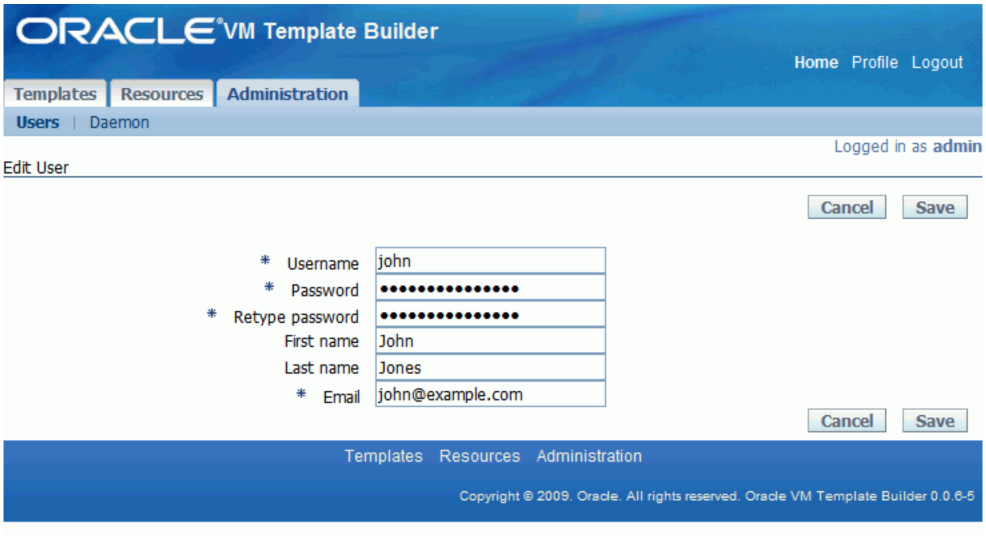
To delete a user, select the user in the list of users and click **Delete**. The user is deleted.

5.1.3 Editing a User

To edit a user:

1. Select the user in the list of users and click **Edit**. The **Edit User** screen is displayed.

Figure 5–3 Edit User Screen



* Username	john
* Password	•••••••••••••••
* Retype password	•••••••••••••••
First name	John
Last name	Jones
* Email	john@example.com

2. Edit any of the following user information:

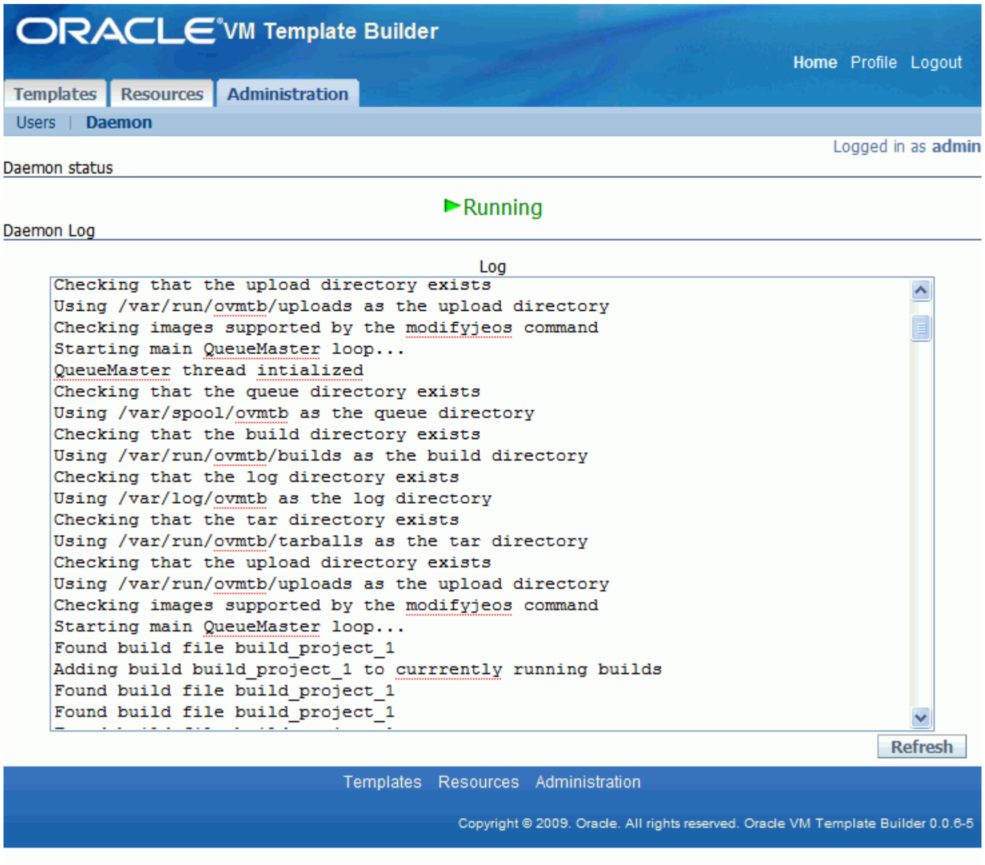
   - **Username**: A name for the user.
   - **Password**: A password for the user.
   - **Retype Password**: Confirm the password by entering it again.
   - **First Name**: The first name of the user.
   - **Last Name**: The last name of the user.
   - **Email**: An email address for the user. This email address is used when resetting the user's password.

   Click **Save**. The **Users** screen is displayed.

5.2 Oracle VM Template Builder Daemon

The **Daemon** sub-tab displays the status of the Oracle VM Template Builder operating system process and displays the process log. To refresh the contents of the page, click **Refresh**. No other operations can be performed on this page.

Figure 5–4 Daemon Status Screen



```
Checking that the upload directory exists
Using /var/run/ovmtb/uploads as the upload directory
Checking images supported by the modifyjeos command
Starting main QueueMaster loop...
QueueMaster thread intialized
Checking that the queue directory exists
Using /var/spool/ovmtb as the queue directory
Checking that the build directory exists
Using /var/run/ovmtb/builds as the build directory
Checking that the log directory exists
Using /var/log/ovmtb as the log directory
Checking that the tar directory exists
Using /var/run/ovmtb/tarballs as the tar directory
Checking that the upload directory exists
Using /var/run/ovmtb/uploads as the upload directory
Checking images supported by the modifyjeos command
Starting main QueueMaster loop...
Found build file build_project_1
Adding build build_project_1 to currrently running builds
Found build file build_project_1
Found build file build_project_1
```

6

Creating Templates

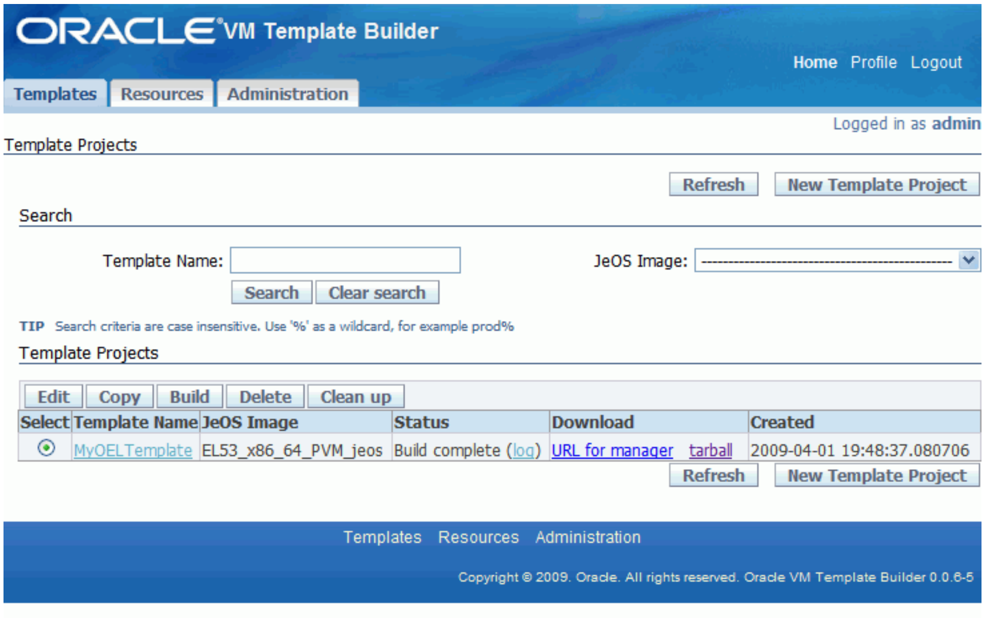
This Chapter discusses creating and managing Oracle VM templates. This Chapter contains:

- Creating Templates
- Building Templates
- Editing Templates
- Copying Templates
- Deleting Templates
- Cleaning Up Templates

To perform the actions discussed in this Chapter, select the **Templates** tab. **The Template Projects** screen is displayed.

The **Template Projects** screen is shown in Figure 6–1, "Template Projects Screen".

Figure 6–1 Template Projects Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin
Template Projects
Refresh New Template Project
Search
Template Name:
JeOS Image:
Search Clear search
Template Projects
Edit Copy Build Delete Clean up
Refresh New Template Project
Templates Resources Administration

TIP Search criteria are case insensitive. Use '%' as a wildcard, for example prod%

Select	Template Name	JeOS Image	Status	Download	Created
●	MyOELTemplate	EL53_x86_64_PVM_jeos	Build complete (log)	URL for manager tarball	2009-04-01 19:48:37.080706

Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

You can create a new template by clicking the **New Template Project** button.

You can search for a template by entering part or all of the template name in the **Template Name** field, and clicking **Search**. Similarly, you can display a list of templates for an operating system by selecting the operating system in the **JeOS Image** drop down and clicking **Search**. Or use both search criteria together to search for a template.

You can manage templates using the buttons at the top left of the **Template Projects** table. The buttons related to this table are:

- **Edit:** Edits a template.
- **Copy:** Copies the selected template and lists it in the **Templates Projects** table.
- **Build:** Builds the selected template.
- **Delete:** Deletes the selected template.
- **Clean Up:** Removes the selected template image file and tarball. The template definition is not deleted.

6.1 Creating Templates

Oracle VM Template Builder uses JeOS to facilitate building an operating system instance with only the absolute minimum packages you need for a template. The operating systems you can select in Oracle VM Template Builder are derived from the available JeOS images. See Section 4.3, "JeOS Images" in Chapter 4, "Managing Resources"for more information on JeOS images.

To create an Oracle VM template, click **New Template Project** on the **Template Projects** screen. A template creation wizard guides you through the steps to create a template. The steps are:

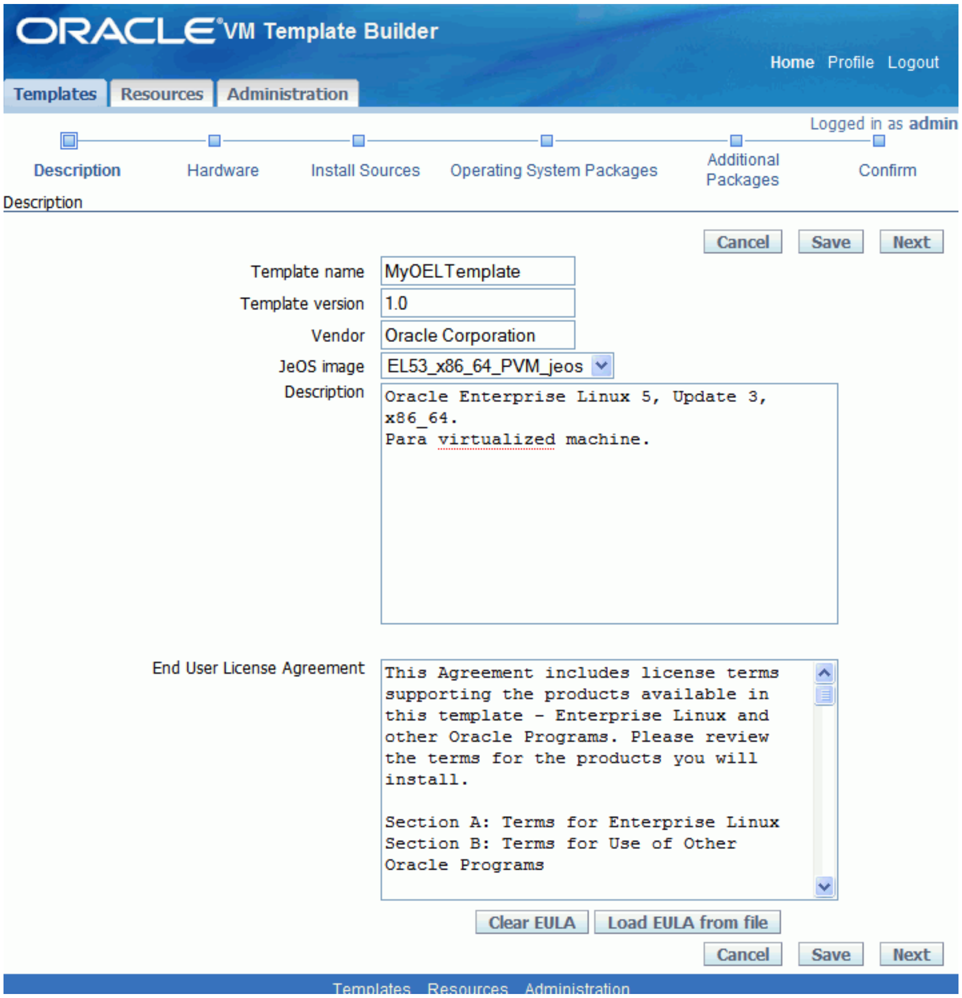
- Adding a Description
- Adding Virtual Machine Hardware
- Setting the Operating System Installation Source
- Selecting Operating System Packages
- Adding Additional Software Packages and Template Script
- Confirming and Building the Template

Each step of the wizard and the fields on each screen are discussed in this section.

6.1.1 Adding a Description

The first step in creating a template is to enter descriptive information about the template in the **Template Description** Screen.

Figure 6–2 Template Description Screen



Template name: MyOELTemplate
Template version: 1.0
Vendor: Oracle Corporation
JeOS image: EL53_x86_64_PVM_jeos
Description:

```
Oracle Enterprise Linux 5, Update 3,
x86_64.
Para virtualized machine.
```

End User License Agreement:

```
This Agreement includes license terms
supporting the products available in
this template - Enterprise Linux and
other Oracle Programs. Please review
the terms for the products you will
install.

Section A: Terms for Enterprise Linux
Section B: Terms for Use of Other
Oracle Programs
```

This section of the wizard gives you fields to enter information to describe the template, manage version control, and manage End User Licence Agreements (EULAs). When you have completed this screen, click **Next** to continue.

Template Name

Enter a name for the template, for example, MyTemplate.

Template Version

Enter a version number for the template.

Vendor

Enter the name of the template vendor, for example, MyCompany.

JeOS Image

Select a JeOS image from the drop down list. This image is used to create the operating system for the guest virtual machine in the template. The list of operating systems is derived from the operating systems supported by JeOS. You cannot add to this list.

Description

Enter a description of the template.

End User License Agreement

If required, enter an End User License Agreement (EULA) to be included in the template.

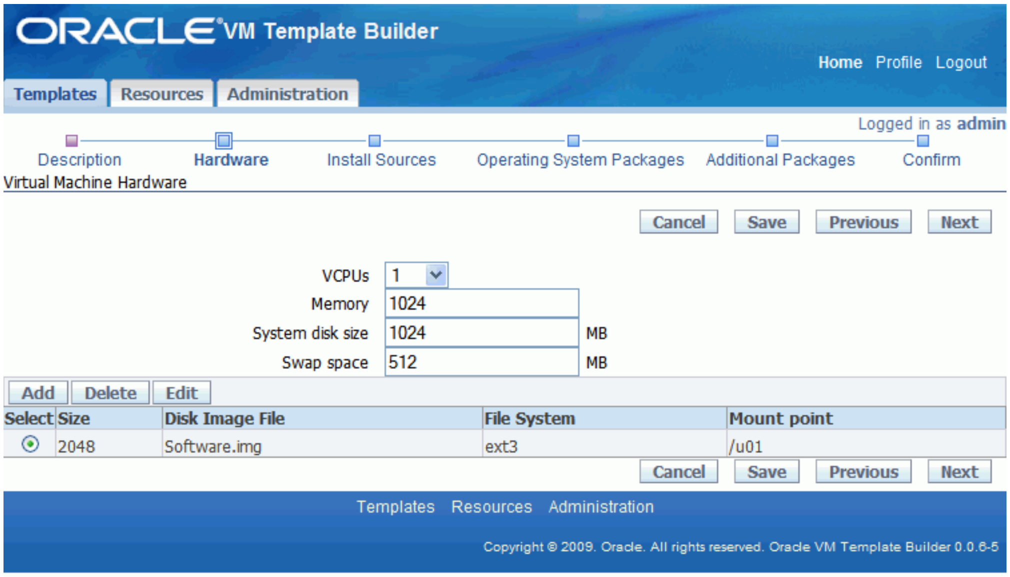
To add an EULA, click **Load EULA from file**. The **Add EULA** screen is displayed. Enter the location for the EULA. The EULA location must be on your local machine. Click **Save** to load the EULA.

To remove an EULA, click **Clear EULA**. The EULA is removed from the template.

6.1.2 Adding Virtual Machine Hardware

The next step in creating a template is to enter the virtual machine hardware information for the template in the **Virtual Machine Hardware** screen.

Figure 6–3 Virtual Machine Hardware Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin
Description
Hardware
Install Sources
Operating System Packages
Additional Packages
Confirm
Virtual Machine Hardware
Cancel Save Previous Next
Add Delete Edit
Cancel Save Previous Next
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

VCPUs	1	
Memory	1024	
System disk size	1024	MB
Swap space	512	MB

Select	Size	Disk Image File	File System	Mount point
●	2048	Software.img	ext3	/u01

When you have completed this screen, click **Next** to continue.

VCPUs

Enter the number of virtual CPUs (VCPUs) for the guest virtual machine in the template.

Memory

Enter the amount of memory, in MBs, for the guest virtual machine in the template.

System Disk Size

Enter the amount, in MBs, of free space to be made available on the system disk, after the installation of the operating system.

Swap Space

Enter the size in MBs for the swap space.

Additional Disks

To add additional disks to the guest virtual machine in the template, click **Add**. A set of fields are displayed to enter information about the additional disk. Enter or select:

- **Size**: The size of the disk in MB, for example, 1024 for 1GB.
- **Mount point**: The mount point for the disk, for example, /u01.
- **Disk image file**: A name for the disk image, for example, Software.img.

Click **Save** to save the disk and return to the **Virtual Machine Hardware** screen. The new disk is listed in the table.

To edit a disk, select the disk in the table and select the **Edit** button. Edit the disk and click **Save** to save the disk. The **Virtual Machine Hardware screen** is displayed.

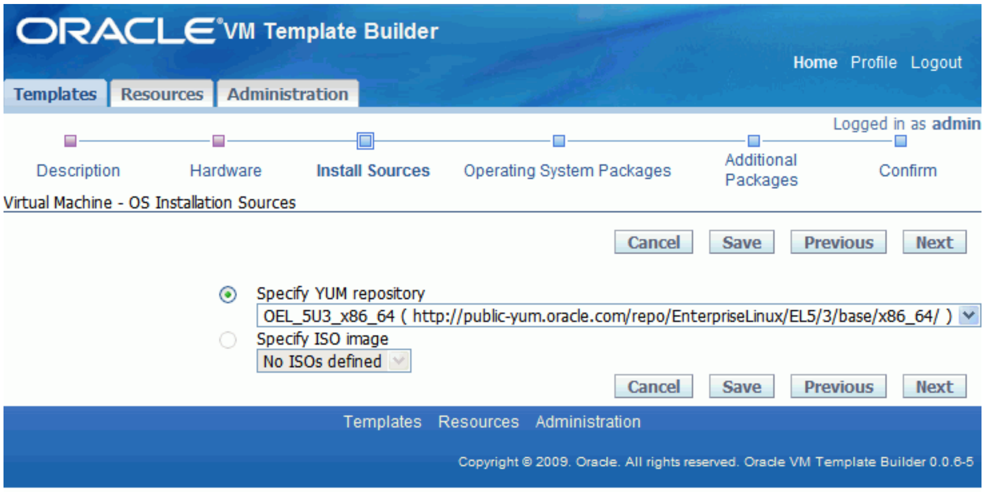
To remove additional disks from the template, select the disk in the table and click **Delete**.

> **Note:** Due to a known issue, only one additional disk image can be added in this release. See Appendix C.4.6, "Only One Additional Disk Is Included In a Template".

6.1.3 Setting the Operating System Installation Source

The next step in creating a template is to enter the virtual operating system installation source for the template in the **Virtual Machine - OS Installation Sources** screen.

Figure 6–4 Virtual Machine - OS Installation Sources



Virtual Machine - OS Installation Sources

● Specify YUM repository

OEL_5U3_x86_64 (http://public-yum.oracle.com/repo/EnterpriseLinux/EL5/3/base/x86_64/)

○ Specify ISO image

No ISOs defined

ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin
Description
Hardware
Install Sources
Operating System Packages
Additional Packages
Confirm
Cancel Save Previous Next
Cancel Save Previous Next
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

This section of the wizard gives you fields to select the location for the operating system installation files to be used for the guest virtual machine in the template. The operating system source may be from a DVD ISO file, or a Yum repository. You can configure the available Yum repositories or DVD ISOs using the **Resources** tab. See Chapter 4, "Managing Resources" for more information on configuring resources. Select one option:

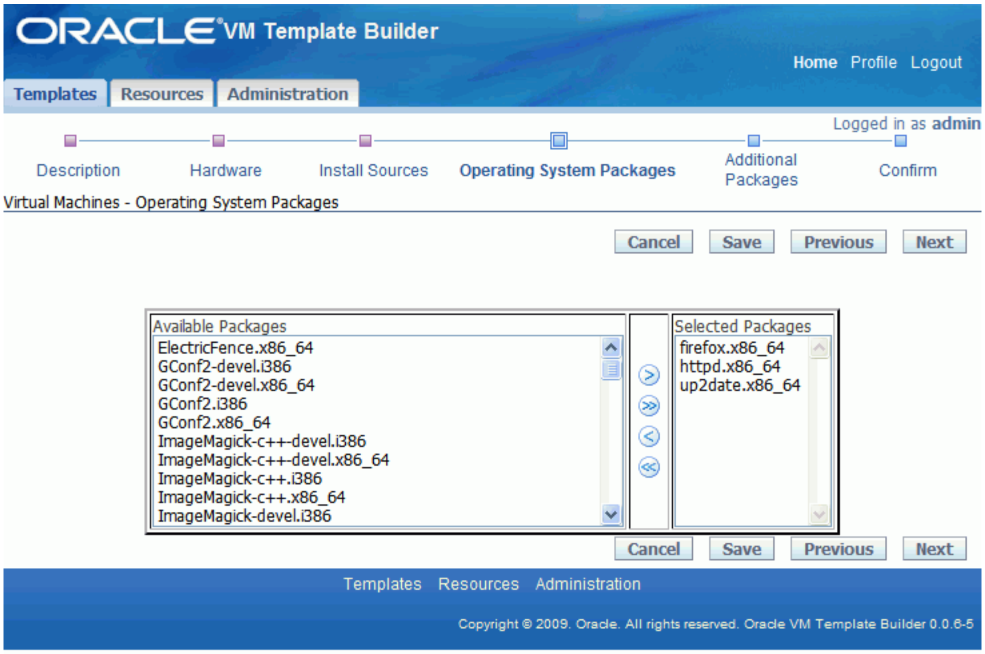
- **Specify Yum repository:** The Yum repository for the operating system.
- **Specify ISO Image**: The DVD ISO image for the operating system.

Click **Next**. The **Virtual Machines - Operating System Packages** screen is displayed.

6.1.4 Selecting Operating System Packages

The next step in creating a template is to select the installation groups and operating system packages to include in the template in the **Virtual Machine - Operating System Packages** screen.

Figure 6–5 Virtual Machine - Operating System Packages Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin
Description
Hardware
Install Sources
Operating System Packages
Additional Packages
Confirm
Cancel Save Previous Next
Cancel Save Previous Next
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

Virtual Machines - Operating System Packages

Available Packages
ElectricFence.x86_64
GConf2-devel.i386
GConf2-devel.x86_64
GConf2.i386
GConf2.x86_64
ImageMagick-c++-devel.i386
ImageMagick-c++-devel.x86_64
ImageMagick-c++.i386
ImageMagick-c++.x86_64
ImageMagick-devel.i386

Selected Packages
firefox.x86_64
httpd.x86_64
up2date.x86_64

This section of the wizard gives you fields to select installation groups and additional software packages from the Yum repository or DVD ISO.

When you have completed this screen, click **Next** to continue

Available Groups

Select the installation group(s) for the guest virtual machine, for example, web-server and authoring-and-publishing. If there are no installation groups included in the Yum repository or DVD ISO, this option is not displayed.

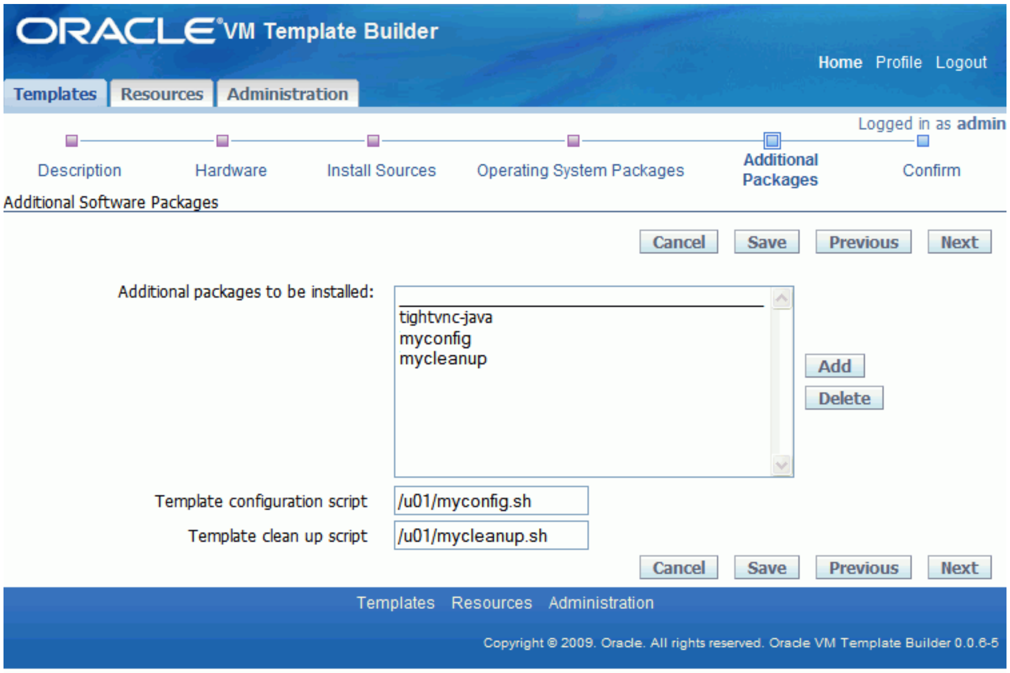
Available Packages

Select any additional operating system packages required for the installation from the **Available Packages** column, for example, firefox. Click the shuttle arrow to move the packages into the **Selected Packages** column. Any dependent packages are not immediately marked to be added, but are installed when the template is built.

6.1.5 Adding Additional Software Packages and Template Script

The next step in creating a template is to upload any additional software packages that are not included in the Yum repository. You can also set template configuration and clean up scripts. These actions are performed on the **Additional Software Packages** screen

Figure 6–6 Additional Software Packages Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin
Description
Hardware
Install Sources
Operating System Packages
Additional Packages
Confirm
Additional Software Packages
Cancel Save Previous Next
Add
Delete
Cancel Save Previous Next
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

Additional packages to be installed: tightvnc-java, myconfig, mycleanup

Template configuration script: /u01/myconfig.sh

Template clean up script: /u01/mycleanup.sh

This section of the wizard enables to you add any additional software packages or applications, for example, the Oracle Database 10*g* Express Edition RPM, or application software configuration and clean up script RPMs. This screen also enables you to add the location of template configuration and clean up scripts in the guest virtual machine created by the template. When you have completed this screen, click **Next** to continue.

Additional Software Packages

To add an RPM, click **Add**. The **Add Package** screen is displayed. Select **URL** and enter an HTTP or FTP location for the RPM, or select **Package file** and enter the location on your local machine for the RPM. Click **Save** to add the RPM and return to the **Additional Software Packages** screen. The RPM is added to the guest virtual machine template.

To delete an RPM, select it in the box and click **Delete**. The RPM is removed from the guest virtual machine template.

Template Scripts

To include your own template configuration script, enter the script location in the **Template configuration script** field, for example, enter /u01/myconfig.sh.

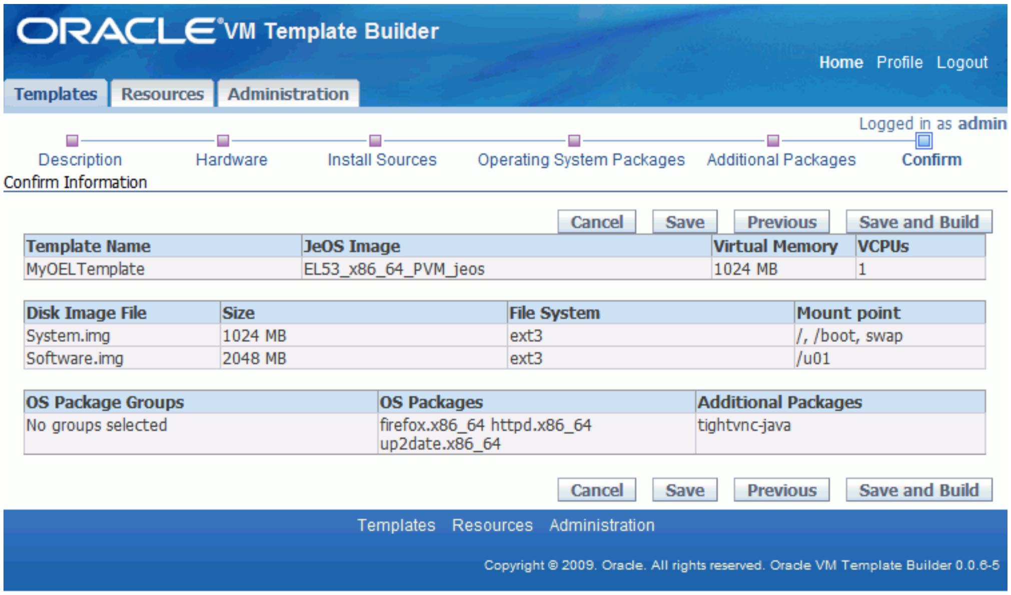
To include your own template clean up script, enter the script location in the **Template clean up script** field, for example, enter /u01/mycleanup.sh.

Both scripts must be on the guest virtual machine running Oracle VM Template Builder. For information on creating template scripts, see Chapter 7, "Creating Template Scripts".

6.1.6 Confirming and Building the Template

The final step in creating a template is to confirm the content and build the template. This is performed on the **Confirm Information** screen.

Figure 6–7 Confirm Information Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin
Description
Hardware
Install Sources
Operating System Packages
Additional Packages
Confirm
Confirm Information
Cancel Save Previous Save and Build
Cancel Save Previous Save and Build
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

Template Name	JeOS Image	Virtual Memory	VCPUs
MyOELTemplate	EL53_x86_64_PVM_jeos	1024 MB	1

Disk Image File	Size	File System	Mount point
System.img	1024 MB	ext3	/, /boot, swap
Software.img	2048 MB	ext3	/u01

OS Package Groups	OS Packages	Additional Packages
No groups selected	firefox.x86_64 httpd.x86_64 up2date.x86_64	tightvnc-java

This section of the wizard enables you to review and confirm the set up of the guest virtual machine to be created in the template, and to initiate a build of the project. A summary of the template is displayed in the **Confirm Information** screen showing:

- The template name
- The JeOS image on which to base the template
- The amount of virtual memory
- The number of virtual CPUs
- The disk(s) to be included in the template, including the disk name, size, file system type and mount point
- The operating system packages to be installed
- The additional packages to be installed

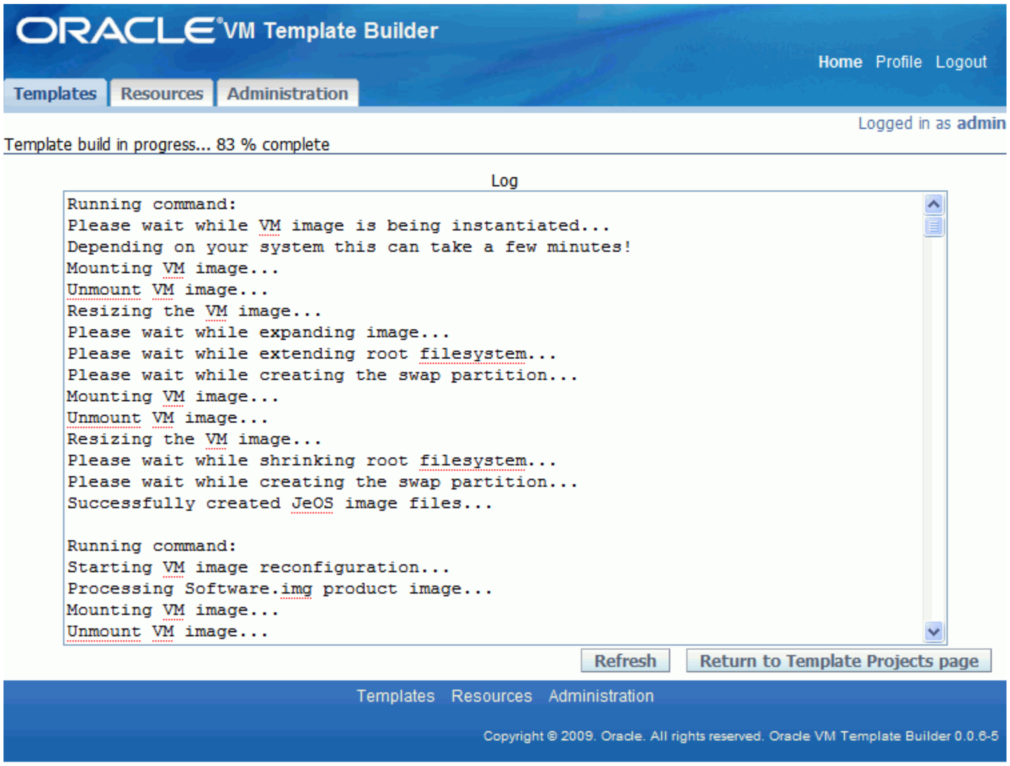
Select **Save** to save the template and return to the **Template Projects** screen. You can build the template from the **Template Projects** screen.

Select **Save and Build** to save and build the template. The **Template build** screen is displayed. Click **Refresh** to display the latest information from the build log. Click **Return to Template Projects page** to return to the Template Projects screen. You can exit the build screen at any time and the build continues.

6.2 Building Templates

To build a template, select the template from the **Template Projects** table on the **Template Projects** screen and click **Build**. The **Template build** screen is displayed and shows the status of the template build process.

Figure 6–8 Template Build Screen


ORACLE VM Template Builder
Home Profile Logout
Templates Resources Administration
Logged in as admin

Template build in progress... 83 % complete

Log

```
Running command:
Please wait while VM image is being instantiated...
Depending on your system this can take a few minutes!
Mounting VM image...
Unmount VM image...
Resizing the VM image...
Please wait while expanding image...
Please wait while extending root filesystem...
Please wait while creating the swap partition...
Mounting VM image...
Unmount VM image...
Resizing the VM image...
Please wait while shrinking root filesystem...
Please wait while creating the swap partition...
Successfully created JeOS image files...

Running command:
Starting VM image reconfiguration...
Processing Software.img product image...
Mounting VM image...
Unmount VM image...
```

Refresh
Return to Template Projects page
Templates Resources Administration
Copyright © 2009. Oracle. All rights reserved. Oracle VM Template Builder 0.0.6-5

Click **Refresh** to display the latest information from the build log. Click **Return to Template Projects page** to return to the Template Projects screen. You can exit the build screen at any time and the build continues.

6.3 Accessing Templates

When you have created and built a template, Oracle VM Template Builder creates an archive (in *.tar.gz* format) of the template, and also creates a directory in which the *system.img* and *vm.cfg* file are located. You can access templates using the links in the **Download** column of the **Template Projects** table on the **Templates** screen.

6.4 Editing Templates

To edit a template, select the template from the **Template Projects** table and click **Edit**. Alternatively, click the link for the template in the **Template Name** column of the **Template Projects** table. The create template wizard is started. Edit the template as required.

6.5 Copying Templates

To copy a template, select the template from the **Template Projects** table and click **Copy**. A copy of the template is displayed in the table.

6.6 Deleting Templates

To delete a template, select the template from the **Template Projects** table and click **Delete**. The template is deleted from the table.

6.7 Cleaning Up Templates

You can remove template images and tarballs you may no longer require. The template definition remains, so you can rebuild the template image and tarball if and when required. Cleaning up templates makes available disk space for new templates to be built.

To clean up a template, select the template from the **Templates Projects** table and click **Clean Up**. The template image and tarball is deleted.

7 Creating Template Scripts

You can create application software configuration and clean up scripts to package with Oracle VM Template Builder templates. Oracle VM Template Builder includes a function library in all templates and provides some standard configuration on templates. The default configuration and function library are a useful starting point for developing application specific scripts.

This Chapter discusses the guest virtual machine configuration processes that occur when a guest virtual machine created from a template starts up, and how to enable application specific configuration with template scripts. This Chapter contains:

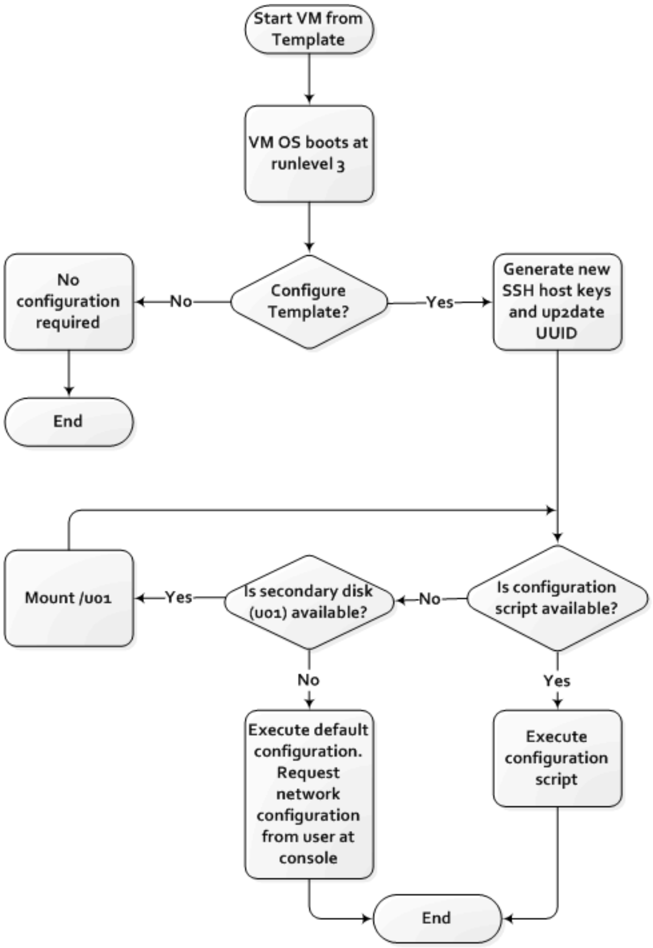
- Configuring the Guest Virtual Machine at Start Up
- Template Configuration Settings
- Template Configuration Scripts
- Template Clean Up Scripts
- Testing Template Scripts
- Configuration Script Function Library

7.1 Configuring the Guest Virtual Machine at Start Up

The ovm-template-config RPM (included with Oracle VM Template Builder) includes a guest virtual machine configuration script, /usr/sbin/oraclevm-template.sh. This script configures a guest virtual machine's operating system if no application-specific scripts are found. If an application-specific script is found, the oraclevm-template script calls that script to perform the operating system and application software configuration. If an application software script is required, the template developer should develop it and package it into the template.

The process of starting a guest virtual machine created from a template, and the subsequent configuration options is shown in Figure 7–1, "Guest virtual machine start up and configuration process".

Figure 7–1 Guest virtual machine start up and configuration process


Start VM from Template
VM OS boots at runlevel 3
Configure Template?
No
No configuration required
End
Yes
Generate new SSH host keys and up2date UUID
Is configuration script available?
No
Is secondary disk (u01) available?
Yes
Mount /u01
No
Execute default configuration. Request network configuration from user at console
Yes
Execute configuration script
End

When a guest virtual machine created from a template is started up, the following script is automatically run:

/etc/rc.d/init.d/oraclevm-template

If it is the first time the guest virtual machine is started, the init.d script calls the following script to perform the configuration:

/usr/sbin/oraclevm-template --config

If it is not the first time the guest virtual machine is started, this script is not called. This way the configuration is performed only once. The RUN_TEMPLATE_CONF parameter in the template configuration file, /etc/sysconfig/oraclevm-template, controls whether the configuration script is run. See Section 7.2, "Template Configuration Settings" for information on the template configuration settings.

The `/usr/sbin/oraclevm-template --config` script either starts an interactive session on the guest virtual machine to configure the operating system, or calls any application software-specific configuration scripts you include with the template. You can run this script on the guest virtual machine at any time. You must be the *root* user to run this script. See "oraclevm-template" in Appendix A, "Command Line Tools" for more information on the oraclevm-template script options.

The template configuration script configures the operating system, not the software installed on it. You can create application software-specific scripts to include in the template, and the template configuration script can call and run these scripts. The template configuration script:

- Regenerates the up2date UUID
- Regenerates the SSH host key
- If an application software configuration script is included, calls and runs this script
- If no application software configuration script is included, calls `/usr/sbin/oraclevm-template --config` to set up the network to use either DHCP or a static IP address

See Section 7.3, "Template Configuration Scripts" for information creating application software configuration scripts.

You can also create a script to perform any post installation configuration, and to clean up logs, temporary files and cache. See Section 7.4, "Template Clean Up Scripts" for information on clean up scripts.

7.2 Template Configuration Settings

There are two important parameters that control the configuration of a guest virtual machine created from a template:

- RUN_TEMPLATE_CONF: Controls whether it is the first time the guest virtual machine is started up, and whether to configure the guest virtual machine.
- TEMPLATE_CONFIG_SCRIPT: Sets the location of any application software script, if required.

These configuration options are set in the template configuration file:

/etc/sysconfig/oraclevm-template

> **Note:** You should not edit this file yourself. This file is updated by Oracle VM Template Builder and by the oraclevm-template command-line interface.

There is also a third parameter in the template configuration file, TEMPLATE_CLEANUP_SCRIPT. This sets the location of any application software clean up script, if required. This parameter is not called during start up of a guest virtual machine created from a template, but can be called from the guest's command line to perform any clean up actions you might need.

7.2.1 RUN_TEMPLATE_CONF

The RUN_TEMPLATE_CONF parameter is the most important, as it controls whether to perform the guest virtual machine configuration. This parameter is set automatically to **YES** when you create a template, and then set to **NO** after a guest virtual machine created from the template is first started up. Setting this parameter to **YES** runs the configuration script and is the default when the template is created. Setting this parameter to **NO** skips the configuration script.

7.2.2 TEMPLATE_CONFIG_SCRIPT

The TEMPLATE_CONFIG_SCRIPT parameter sets the location of the application software configuration script.

This parameter is automatically set in the configuration file when you create a template using the location you set in the **Template configuration script** field on the **Additional Packages** screen. If you do not enter a script location in the template creation wizard, Oracle VM Template Builder uses the default value:

/u01/oraclevm-template.sh

The default location for the application software script is not the system disk, but a separate (second) disk (/u01). If you do not include a separate disk, and do not enter a location on the system disk, this parameter is ignored.

Note: There is no default script provided in this location. It must be provided by the template developer, if required. If you do not provide a script in this location, then no application specific configuration is performed.

7.2.3 TEMPLATE_CLEANUP_SCRIPT

The TEMPLATE_CLEANUP_SCRIPT parameter sets the location of the template clean up script. This parameter is not required for template deployment, but may be useful to edit or clean up the guest virtual machine after a guest has been configured.

This parameter is automatically set in the configuration file when you create a template using the location you set in the **Template clean up script** field on the **Additional Packages** screen. If you do not enter a script location in the template creation wizard, Oracle VM Template Builder uses the default value:

/u01/template-cleanup.sh

The default location for the clean up script is not the system disk, but a separate (second) disk (/u01). If you do not include a separate disk, and do not enter a location on the system disk, this parameter is ignored.

Note: There is no default script provided in this location. It must be provided by the template developer, if required. If you do not provide a script in this location, then no application specific clean up is performed.

7.3 Template Configuration Scripts

You can create your own scripts to configure the software applications in a guest virtual machine created by a template. These scripts can perform any software configuration, such as setting environment variables, creating users, running other scripts, editing system files, starting services, and so on. You can include the script in a template with the template creation wizard.

Application software configuration scripts must be packaged as part of an RPM and added to the template by the template developer. The RPM can be added to the template in the **Additional Packages** screen in the template creation wizard. The RPM is then installed when the operating system is created in the guest virtual machine created from the template. To make sure the script is run when the guest virtual machine is first started, you should include the full path to the configuration script in the **Template configuration script** field on the **Additional Packages** screen. This is the location of the script when it is installed by the RPM on the guest virtual machine created from the template.

Note: Creating and packaging an RPM is beyond the scope of this book.

A log is created in the guest virtual machine created by the template to output any errors or messages when these scripts are run. The log is located at:

/var/log/oraclevm-template.log

There are some template script functions that you can call from an existing function library located on Oracle VM Template Builder guest virtual machine.

Section 7.6, "Configuration Script Function Library" lists the functions you can call in your configuration scripts. Examples of calling and using the functions are included in Appendix B, "Template Script Examples".

7.4 Template Clean Up Scripts

You can create your own script to clean up the operating system and applications in a guest virtual machine created from a template. A clean up script may be useful to reset the guest virtual machine to its original state (before running any configuration script), so you can run the configuration script again. This might be useful in a template testing scenario. You may also want to remove any temporary files, or clear any cache.

Clean up scripts must also be packaged as part of an RPM and added to the template by the template developer. The RPM can be added to the template in the **Additional Packages** screen in the template creation wizard. The RPM is then installed when the operating system is created in the guest virtual machine created from the template. You should include the full path to the clean up script in the **Template clean up script** field on the **Additional Packages** screen. This is the location of the script when it is installed by the RPM on the guest virtual machine created from the template.

The clean up script is not automatically run when a guest virtual machine is created from a template. You must manually run your clean up script in the guest virtual machine using the command:

```
# /usr/sbin/oraclevm-template --cleanup
```

7.5 Testing Template Scripts

It is often useful to test your template configuration scripts as you develop them. To do this, you should deploy the template and start the guest virtual machine. The first time the guest is started the template configuration script is run. To run the configuration script again, you must set the template configuration file to enable the script to run (that is, set RUN_TEMPLATE_CONF to YES):

```
# /usr/sbin/oraclevm-template --enable
```

You can then run the template configuration script using:

```
# /usr/sbin/oraclevm-template --config
```

To perform any operating system or application software clean up, such as resetting variables, removing temporary files or cache, you can use a clean up script. This script is also configured in the template configuration file. You can run the clean up script using:

```
# /usr/sbin/oraclevm-template --cleanup
```

The clean up script is not run automatically when the guest virtual machine is started. It must always be run manually.

The full list of parameters for the template configuration script included in "oraclevm-template" in Appendix A, "Command Line Tools".

7.6 Configuration Script Function Library

There are some functions available in the configuration script function library installed on all guest virtual machines created from templates. You can call these functions in your configuration and clean up scripts. The function library is located at:

/usr/lib/oraclevm-template/functions

This section lists the functions you can call in your application software and clean up scripts. Examples of scripts using these functions are given in Appendix B, "Template Script Examples".

ovm_configure_network [dhcp [*hostname*]] | [static] | [*ip_address netmask gateway dns_server hostname*]
Sets the guest virtual machine network configuration options.

Setting this parameter to **dhcp** configures the guest virtual machine to use DHCP. You can set a host name with the ***hostname*** parameter, for example:

```
ovm_configure_network "dhcp"
```

or

```
ovm_configure_network "dhcp" "myhost.example.com"
```

Setting this parameter to **static** configures the guest virtual machine to use a static IP address. You are prompted to enter the static IP information during the initial guest start up, for example:

```
ovm_configure_network "static"
```

To set a static IP address, enter the ***ip_address***, ***netmask***, ***gateway***, ***dns_server***, and ***hostname***.

```
ovm_configure_network "192.168.2.100" "255.255.255.0" "192.168.2.1" "192.168.2.1"
"myhost.example.com"
```

If you do not enter any parameters, you are prompted to use either DHCP or enter a static IP address when the guest virtual machine starts up for the first time.

ovm_cleanup_os
Cleans up the guest's operating system and resets the network to use DHCP. For a full list of the actions performed by this function, see the --cleanup option in the oraclevm-template command in Appendix A, "Command Line Tools".

ovm_disable_firewall
Disables the firewall on the guest virtual machine.

press_anykey
Pauses execution of the script and waits for the user the press any key before continuing.

ovm_error *message*
Echoes ***message*** to the screen and writes it to the log file.

ovm_info *message*
Echoes *message* to the screen and writes it to the log file.

ovm_warn *message*
Echoes *message* to the screen and writes it to the log file.

8 Deploying Templates

This Chapter describes the Oracle VM template deployment process. This Chapter contains:

8.1 Template Format and Location

When you have created and built a template, Oracle VM Template Builder creates an archive (in *.tar.gz* format) of the template and saves it on the Oracle VM Template Builder guest virtual machine. You can access templates using the links in the **Download** column of the **Template Projects** table on the **Templates** screen. The links to the templates are in the format:

http://*host*/ovmtb/data/tarballs/project_*ID*.tar.gz

Oracle VM Template Builder also creates a directory in which the *system.img* and *vm.cfg* file are located:

http://*host*/ovmtb/data/builds/project_*ID*/

8.2 Deploying Templates

To deploy an Oracle VM template using Oracle VM Manager, either:

- Copy or move the template *.tar* file to an Oracle VM Server. The *.tar* file location is provided on the **Template Projects** screen in Oracle VM Template Builder. Untar the template and import it using the **Discover and register** import option in Oracle VM Manager.
- Import the template into Oracle VM Manager using the URL provided on the **Template Projects** screen in Oracle VM Template Builder using the **HTTP or FTP** import option.

For more information on deploying templates in Oracle VM Manager, see the *Oracle VM Manager User's Guide*.

8.3 Guest Virtual Machine Configuration

When you start a guest virtual machine for the first time, you are prompted to enter information about the guest's network configuration. The following configuration questions are displayed:

```
Do you want to enable dynamic IP configuration (DHCP) (Y|n)
```

Enter `n` and press **Enter** to manually assign an IP address.

Enter `Y` and Press **Enter** to enable DHCP and have an IP address automatically assigned.

If you select to use DHCP, the following question is displayed:

```
Do you want to manually configure the hostname (y|n)? [n]
```

Enter `y` to enter a host name, and `n` if you do not want to enter a host name, and press **Enter**. If you entered `y`, the following question is displayed:

```
Enter hostname (e.g. host.domain.com):
```

Enter a host name and press **Enter**.

If you selected to manually assign an IP address, you are prompted to enter the IP address, netmask, gateway and DNS server:

```
Enter static IP address:
Enter netmask
Enter gateway:
Enter DNS server:
```

Enter your network information and press **Enter** after each entry. You are prompted to enter a host name for the guest virtual machine:

```
Enter hostname (e.g. host.domain.com):
```

Enter a host name and press **Enter**.

The guest virtual machine configuration is complete.

The operating system login prompt is displayed.

Note: The default operating system username and password for guest virtual machines created from templates built with Oracle VM Template Builder is:

Username: root

Password: ovsroot

A

Command Line Tools

This Appendix contains reference information for Oracle VM Template Builder command line tools and scripts, and contains:

- oraclevm-template

oraclevm-template

The ovm-template-config RPM (included with Oracle VM Template Builder) includes a template configuration script, oraclevm-template. This script configures a template's operating system, and calls application software configuration scripts. When a guest virtual machine created from a template is started, the following script is automatically run:

/etc/rc.d/init.d/oraclevm-template

This script calls the /usr/sbin/oraclevm-template script with the `--config` parameter to perform the operating system, and application software configuration.

You can run this script at any time as the root user. The syntax for this script is:

```
/usr/sbin/oraclevm-template [option]
```

Options

--config
Performs the following configuration on the guest virtual machine:

- Regenerates the up2date UUID
- Regenerates the SSH host key
- Runs the template configuration script if it is specified in the TEMPLATE_CONFIG_SCRIPT parameter in the template configuration file, and found on the guest virtual machine. If no template configuration script is specified or found, this parameter sets up the network to use either DHCP or a static IP address by requesting this information from the user at the command console.

--enable
Runs the template configuration script the next time the guest virtual machine is started. Sets the RUN_TEMPLATE_CONF parameter in the template configuration file to **YES**.

--disable
Disables the template configuration script the next time the guest virtual machine is started. Sets the RUN_TEMPLATE_CONF parameter in the template configuration file to **NO**.

--status
Displays whether the template configuration script runs the next the time the guest virtual machine is started.

--cleanup
Performs the following clean up actions on the guest virtual machine:

- If a clean up script is specified in the TEMPLATE_CLEANUP_SCRIPT parameter in the template configuration file, and the script is found, runs this script
- Cleans up the operating system log files in /var/log
- Deletes the systemid from the up2date configuration file
- Deletes the yum cache
- Deletes any DHCP client cache

- Deletes the root user's SSH configuration file and bash history files
- Deletes the /etc/resolv.conf file
- Resets the /etc/hostnames file to the default
- Resets the network configuration to DHCP
- Shuts down the network service

B

Template Script Examples

This Appendix contains examples of Oracle VM Template Builder configuration and cleanup scripts, and contains:

- Oracle Database 11g Configuration Script
- Oracle Database 11g Clean Up Script
- Oracle Enterprise Manager Agent Configuration Script
- Oracle Enterprise Manager Agent Clean Up Script
- Oracle VM Template Builder Configuration Script

B.1 Oracle Database 11*g* Configuration Script

This example configuration script is used in the Oracle Database 11*g* templates.

```
#!/bin/sh
#
# functions
fail () {
    echo
    ovm_error "$*"
    ovm_error "Failed to install Oracle Database."
    press_anykey
    exit 1
}

relink_binaries () {
    echo
    ovm_info "Relinking Oracle Binaries..."
    su -s /bin/bash oracle -c \
        "export ORACLE_HOME=$ORACLE_HOME; \
         cd $ORACLE_HOME/bin && ./relink all >/home/oracle/relink.log 2>&1"

    if [ `grep -ic -e "undefined reference" \
          -e " failed " /home/oracle/relink.log` -gt 0 ]; then
        ovm_error "Oracle Relinking Failed. Logs: /home/oracle/relink.log"
        echo "To build again do :"
        echo "cd $ORACLE_HOME/bin "
        echo "./relink all "
        fail "relink error."
    else
        ovm_info "Oracle Relinking Completed Successfully"
        ovm_info "Logs: /home/oracle/relink.log"
    fi
}

configure_ask()
{

    cat <<EOF

Oracle Database Configuration
-------------------------------------------------
This will configure on-boot properties of Oracle Database.
The following questions will determine whether the database should be
starting upon system boot, the ports it will use, and the passwords that
will be used for database accounts.  Press <Enter> to accept the defaults.

EOF

    # prompt for relink or not
    while true; do
        echo -n "Do you want to relink binaries? (y/n) [n] " && read n
        case "$n" in
            y|Y|yes|Yes|YES)
                DO_RELINK=true
                break;;
            ""|n|N|no|No|NO)
                DO_RELINK=false
                break;;
            *)  continue;;
        esac
    done
    echo

    #get the http port value
    while true; do
        while true; do
            echo -n Specify the HTTP port that will be used for Oracle Application
Express [8080]:
            read LINE
            if [ -z $LINE ]; then
                LINE=8080
            fi
            port=`netstat -n --tcp --listen | grep :$LINE | awk '{print $4}' | cut
-d':' -f2`
            if [ "$port" = "$LINE" ]; then
                echo Port $port appears to be in use by another application.\
                Please specify a different port.
            else
                break;
            fi
        done
        case "$LINE" in
"")
            break
            ;;
        *[^0-9]*)
            echo "Invalid http port: $LINE"
            ;;
        *)
            HTTP_PORT=$LINE
            break
            ;;
```

```
       esac
    done

    #get the listener port value
    while true; do
        echo
        while true; do
            echo -n Specify a port that will be used for the database listener
[1521]:
            read LINE
            if [ -z $LINE ]; then
                LINE=1521
            fi
            echo
            port=`netstat -n --tcp --listen | grep :$LINE | awk '{print $4}' | cut
-d':' -f2`
            if [ "$port" = "$LINE" ]; then
                echo Port $port appears to be in use by another application.\
                Please specify a different port.
            else
                break;
            fi
        done

        case "$LINE" in
        "")
            break
            ;;
        *[^0-9]*)
            echo "Invalid port: $LINE" >&2
            ;;
    *)
            if [ "$HTTP_PORT" != "$LINE" ]
        then
            LISTENER_PORT=$LINE
                    break
        else
            echo Database listener cannot be configured on the same port as APEX.
        fi
            ;;
        esac
    done

    #get the database password
    while true; do
    echo -n "Specify a password to be used for database accounts.  Note that the
same
password will be used for SYS, SYSTEM and ADMIN for APEX.  Oracle recommends
the use of different passwords for each database account.  This can be done
after initial configuration:"
       while true; do
           /bin/stty -echo > /dev/null 2>&1
   temp=`echo $IFS`
           export IFS="\n"
   while true; do
       read LINE
               while [ -z "$LINE" ]; do
   echo
   echo -n "Password can't be null. Enter password:"
           read LINE
       done
       result=`expr index "$LINE" [\'\"]`
       if [ $result != 0 ]; then
                echo
           echo -n "The password you entered contains invalid characters. Enter 
password:"
       else
                   break
               fi
   done
           echo
           echo -n "Confirm the password:"
           read LINE1
           echo
           if [ "$LINE" != "$LINE1" ]; then
               echo
               echo -n "Passwords do not match.  Enter the password:"
           else
               break
           fi
       done
       /bin/stty echo > /dev/null 2>&1
       ORACLE_PASSWORD=$LINE
       export IFS=$temp
       break;
    done
    # get option enable or not
    while true; do
        if [ "$ORACLE_DBENABLED" = "true" ]; then
            CUR=y
        else
            CUR=n
        fi
        echo
        echo -n "Do you want Oracle Database to be started on boot (y/n) [y]:"
        read LINE
        if [ -z $LINE ]; then
ORACLE_DBENABLED=true
fi
        echo
        case "$LINE" in
        "")
            break
            ;;
        y|Y)
            ORACLE_DBENABLED=true
            break
            ;;
        n|N)
            ORACLE_DBENABLED=false
            break
            ;;
        *)
            echo "Invalid response: $LINE " >&2
            break
        esac
    done
}

configure_perform () {
    ret=0

    # replace the strings
    sed -e "s/%hostname%/$(hostname)/g" \
        -e "s/%port%/$LISTENER_PORT/g" \
        $ORACLE_CONFIG_HOME/listener.ora > $ORACLE_HOME/network/admin/listener.ora

    sed -e "s/%hostname%/$(hostname)/g" \
        -e "s/%port%/$LISTENER_PORT/g" \
        $ORACLE_CONFIG_HOME/tnsnames.ora > $ORACLE_HOME/network/admin/tnsnames.ora

    sed -e "s/%hostname%/$(hostname)/g" \
        -e "s/%port%/$LISTENER_PORT/g" \
        $ORACLE_CONFIG_HOME/initorcl.ora > $ORACLE_HOME/config/initorcl.ora

    chown oracle:dba $ORACLE_HOME/network/admin/listener.ora \
                     $ORACLE_HOME/network/admin/tnsnames.ora \
                     $ORACLE_HOME/config/initorcl.ora

    export ORACLE_HOME
    #start listener
    su -s /bin/bash $ORACLE_OWNER -c "$ORACLE_HOME/bin/lsnrctl start" > /dev/null
2>&1
    # start ASM instance
    # wait for cssd automatically startup. 30 secs.
    echo "Waiting for CSS available."
    ct=0
    while ! $ORACLE_HOME/bin/crsctl check css >/dev/null 2>&1; do
        sleep 3
        ct=$(($ct+1))
        [ $ct -gt 10 ] && break
    done
    if ! $ORACLE_HOME/bin/crsctl check css >/dev/null 2>&1; then
        echo "Starting Cluster Synchronization Services."
        /etc/init.d/init.cssd start; init q
        # 30 tries with 10 secs interval => 5 mins timeout
        echo "Waiting for CSS available."
        ct=0
        while ! $ORACLE_HOME/bin/crsctl check css >/dev/null 2>&1; do
            sleep 10
            ct=$(($ct+1))
            if [ $ct -gt 30 ]; then
                fail "Failed to start CSS."
            fi
        done
    fi
    if [ "$ORACLE_RNAME" = oracle10g ]; then
        su -s /bin/bash $ORACLE_OWNER -c "ORACLE_SID=+ASM $ORACLE_HOME/bin/sqlplus
-s /nolog" <<SQL
connect / as sysdba
startup
quit
SQL
    ret=$?
    elif [ "$ORACLE_RNAME" = oracle11g ]; then
        su -s /bin/bash  $ORACLE_OWNER -c "ORACLE_SID=+ASM $ORACLE_
HOME/bin/sqlplus -s /nolog" <<SQL
connect / as sysasm
startup
quit
```

```
SQL
    ret=$?
    else
        fail "Unknown database."
    fi
    if [ $ret -ne 0 ]; then
        fail "Failed to start up ASM instance."
    fi

    # run post creation sql
    # 1. create spfile with configured pfile
    # 2. change sys/system password.
    # 3. change APEX ADMIN password
    # 4. enable XML DB HTTP Service
    su -s /bin/bash  $ORACLE_OWNER -c "ORACLE_SID=orcl $ORACLE_HOME/bin/sqlplus -s
/nolog" <<SQL
connect / as sysdba
startup nomount pfile="$ORACLE_HOME/config/initorcl.ora";
create spfile='+DATA/orcl/spfileorcl.ora' FROM pfile='$ORACLE_
HOME/config/initorcl.ora';
alter database mount;
alter database open;
shutdown immediate;
startup;

alter user sys identified by "$ORACLE_PASSWORD";
alter user system identified by "$ORACLE_PASSWORD";

ALTER USER ANONYMOUS ACCOUNT UNLOCK;
ALTER SYSTEM SET SHARED_SERVERS = 5 SCOPE=BOTH;

@$ORACLE_HOME/apex/apxxepwd "$ORACLE_PASSWORD";
begin
    dbms_xdb.sethttpport('$HTTP_PORT');
    dbms_xdb.setftpport('0');
end;
/
commit;

exec dbms_xdb.cfg_refresh;
quit;
SQL
    if [ $? -ne 0 ]; then
       fail "Failed to run post database creation script."
    fi
   # open http port and database port, sshd port as well
   system-config-securitylevel-tui --quiet \
       --port=$LISTENER_PORT \
       --port=$HTTP_PORT \
       --port=ssh

   # add ASM and database instances to /etc/oratab
    if [ -f "$ORATAB" ]; then
        if grep -q "^+ASM:$ORACLE_HOME" $ORATAB ; then
            sed -i "s|+ASM:$ORACLE_HOME:.|+ASM:$ORACLE_HOME:Y|g" $ORATAB
        else
            echo "+ASM:$ORACLE_HOME:Y" >> $ORATAB
        fi
        if grep -q "^orcl" $ORATAB; then
            sed -i "s|orcl:$ORACLE_HOME:.|orcl:$ORACLE_HOME:W|g" $ORATAB
        else
            echo "orcl:$ORACLE_HOME:W" >> $ORATAB
        fi
    else
        echo "+ASM:$ORACLE_HOME:Y" >> $ORATAB
        echo "orcl:$ORACLE_HOME:W" >> $ORATAB
        chown oracle:root $ORATAB
    fi
    return 0
}

write_sysconfig()
{
    cat >"$CONFIGURATION" <<EOF
#This is a configuration file for automatic starting of the Oracle
#Database and listener at system startup.

# ORACLE_DBENABLED:'true' means to load the Database at system boot.
ORACLE_DBENABLED=${ORACLE_DBENABLED:-false}
EOF
    ovm_log "Wrote to $CONFIGURATION"
}

# tmpfs check only for 11g
check_tmpfs() {
    tmpfs_name=tmpfs
    tmpfs_size=0
    tmpfs_used_size=0
    if df | grep -q /dev/shm ; then
       tmpfs_name=$(df | grep /dev/shm | awk '{print $1}')
       tmpfs_size=$(df | grep /dev/shm | awk '{print $2}')
       tmpfs_used_size=$(df | grep /dev/shm | awk '{print $3}')
    fi
    [ $tmpfs_size -ge 2097152 ] && return 0
    ovm_log "set 2GB tmpfs"
    sed -i '/tmpfs/s/defaults/defaults,size=2g/' /etc/fstab
    [ $tmpfs_used_size -eq 0 ] && umount /dev/shm >/dev/null 2>&1
    mount -t tmpfs $tmpfs_name -o size=2g /dev/shm
}
############################################################################

case "$ORACLE_TRACE" in
  T) set -x;;
  *) ;;
esac
# source functions from ovm-template-config package .
/usr/lib/oraclevm-template/functions
. /usr/lib/oraclevm-template/functions


# oracle databse env variables
. /u01/db-config

# display script version
if [ "$1" = "-v" -o "$1" = "--version" ]; then
    echo $VERSION
    exit 0
fi

#######################################################################
```

```
# prerequisites check

# check if oracle-validated has been installed
if ! rpm -q oracle-validated >/dev/null 2>&1; then
    fail "oracle-validated not installed."
fi
# check if required packages have been installed
if ! rpm -q oracleasm-`uname -r` >/dev/null 2>&1; then
    fail "oracleasm-`uname -r` not installed."
fi
if ! rpm -q oracleasmlib >/dev/null 2>&1; then
    fail "oracleasmlib not installed."
fi

# configure and enable asmlib
/etc/init.d/oracleasm configure >/dev/null <<-EOF
oracle
dba
y
y
EOF

# check if asm disk available
if ! /etc/init.d/oracleasm listdisks | grep -q VOL1; then
    fail "ASM disk 'VOL1' not exist."
fi
if ! /etc/init.d/oracleasm listdisks | grep -q VOL2; then
    fail "ASM disk 'VOL2' not exist."
fi
# prerequistes check done
########################################################################

# configure network DHCP/static IP
ovm_configure_network

########################################################################
# start db reconfiguration
ovm_info "Starting Oracle database reconfiguration."

# get configuration from user input
configure_ask
write_sysconfig

if [ "$DO_RELINK" = true ]; then
    relink_binaries
else
    ovm_log "Relinking skipped."
fi

# run orainstRootsh and root.sh
echo
ovm_info "Running orainstRoot.sh and root.sh..."
sed -i 's/localconfig reset/localconfig add/g' $ORACLE_HOME/root.sh
$ORACLE_INVENTORY/orainstRoot.sh
$ORACLE_HOME/root.sh

# check/set tmpfs only for 11g
if [ "$ORACLE_RNAME" = 'oracle11g' ]; then
    # check/re-configure tmpfs size
    check_tmpfs
```

```
fi

# configure and start css
# localconfig is hacked to fix bug, see Note:264235.1
if ! $ORACLE_HOME/bin/crsctl check css >/dev/null 2>&1; then
    echo
    ovm_info "Starting CSS."
    $ORACLE_HOME/bin/localconfig add >/dev/null 2>&1
fi

# reconfiguration
configure_perform

# startup init script
[ ! -e /etc/init.d/dbstart ] && \
    ln -sf "$ORACLE_INIT_PATH" /etc/init.d/dbstart

# enable dbstart service
if ! chkconfig --list | grep -q dbstart; then
    ovm_log "Enabling Oracle Database service"
    chkconfig --add dbstart
    chkconfig --level 35 dbstart on
fi

if [ "$ORACLE_DBENABLED" != true ]; then
    /etc/init.d/dbstart stop
fi

# set env variables for oracle user
cat >> $ORACLE_PROFILE <<-EOF
# variables for oracle database environment
ORACLE_BASE=$ORACLE_BASE
ORACLE_HOME=$ORACLE_HOME
ORACLE_SID=orcl
SAVE_LLP=$LD_LIBRARY_PATH
LD_LIBRARY_PATH=\$ORACLE_HOME/lib:\$SAVE_LLP
PATH=\$PATH:\$ORACLE_HOME/bin/
export ORACLE_BASE ORACLE_HOME ORACLE_SID LD_LIBRARY_PATH PATH
EOF

echo
echo
ovm_info "Configuration Completed Successfully."
# if get a invalid hostname, then give IP address.
host=$(hostname)
if [ "$host" = "localhost.localdomain" ]; then
  host=$(ifconfig eth0 | grep "inet addr" | sed 's/.*inet addr:\([^ ]*\).*/\1/')
fi
ovm_info "To access the Oracle Application Express go to \"http://$host:$HTTP_
PORT/apex\""

press_anykey
```

B.2 Oracle Database 11*g* Clean Up Script

This example clean up script is used in the Oracle Database 11*g* template.

```
#!/bin/sh
#
# functions
```

```
fail () {
    echo
    ovm_error "$*"
    ovm_error "Failed to install Oracle Database."
    press_anykey
    exit 1
}

case "$ORACLE_TRACE" in
  T) set -x;;
  *) ;;
esac

# source functions from ovm-template-config package .
/usr/lib/oraclevm-template/functions
. /usr/lib/oraclevm-template/functions

# oracle databse env variables
. /u01/db-config

# display script version
if [ "$1" = "-v" -o "$1" = "--version" ]; then
    echo $VERSION
    exit 0
fi


########################################################################
# start db reconfiguration
ovm_info "Cleanup Oracle database reconfiguration."

service network restart

/etc/rc.d/init.d/dbstart stop

$ORACLE_HOME/bin/localconfig delete

chkconfig dbstart off
/bin/rm -f /etc/rc.d/init.d/dbstart
/bin/rm -rf /etc/ora*
/bin/rm -f /etc/sysconfig/$ORACLE_RNAME
/bin/rm -rf $ORACLE_HOME/localhost.localdomain_orcl*
/bin/rm -rf $ORACLE_BASE/diag/rdbms/orcl
if [ "$ORACLE_RNAME" = oracle10g ]; then
    /bin/rm -f $ORACLE_BASE/admin/orcl/bdump/* $ORACLE_BASE/admin/orcl/udump/*
$ORACLE_BASE/admin/orcl/adump/*
    /bin/rm -f $ORACLE_BASE/admin/+ASM/bdump/* $ORACLE_BASE/admin/+ASM/udump/*
fi
/bin/rm -f /usr/local/bin/*
sed -i '/database env/,$d' /home/oracle/.bash_profile
/bin/rm -f /home/oracle/relink.log
cat /dev/null > /home/oracle/.bash_history

ovm_cleanup_os
```

B.3 Oracle Enterprise Manager Agent Configuration Script

This example configuration script is used in the Oracle Database 11*g* templates.

```
#!/bin/bash
```

```
case "$ORACLE_TRACE" in
    T) set -x ;;
    *) ;;
esac

# oms_connection function interactively collects user's input for
# OMS hostnam, ip, and port values
oms_connection () {

  local oms_hostname
  local oms_ip
  local oms_host
  local oms_port

  # ask for oms connection infomation.
  echo
  echo "Provide Management Service hostname that the Management Agent will
communicate with."
  while true; do
    # oms hostname
    while true; do
        echo -n "Enter OMS hostname: "
        read oms_hostname
        if [ -z $oms_hostname ]; then
            continue
        fi
        break
    done

    # if can not resolve the hostname, then require IP address
    if ! ping -c 3 $oms_hostname >/dev/null 2>&1; then
        echo "*Can not resolve hostname $oms_hostname."
        while true; do
            echo -n "Enter OMS IP address: "
            read oms_ip
            if [ -z $oms_ip ]; then
                continue
            fi
            break
        done
    fi
    if [ ! -z $oms_ip ]; then
        oms_host=$oms_ip
    else
        oms_host=$oms_hostname
    fi
    # oms port
    echo -n "Enter OMS port: [4889] "
    read oms_port
    if [ -z "$oms_port" ]; then
        oms_port=4889
    fi
    # test connection
    if ! nc -w 3 -z $oms_host $oms_port >/dev/null; then
      echo "*Can NOT connect to $oms_host:$oms_port, please enter OMS information
again."
      oms_ip=
      continue
    else
```

```
      OMS_HOST=$oms_hostname
      OMS_PORT=$oms_port
      if [ ! -z "$oms_ip" ]; then
        /bin/cp -f /etc/hosts /etc/hosts.orabak$$
        sed -i "/^$oms_ip/d" /etc/hosts
        echo "$oms_ip $OMS_HOST ${OMS_HOST%%.*}" >> /etc/hosts
      fi
    fi
    break
  done

  # enter password
  echo
  echo "Provide the Agent Registration password so that the Management Agent can
communicate with Secure Management Service."
  while true; do
    echo -n "Enter Agent Registration Password: "
    stty -echo
    read secure_passwd
    stty echo
    echo
    if [ -z $secure_passwd ]; then
      continue
    fi
    break
  done
  SECURE_PASSWD=$secure_passwd
}

# source functions that are part of standard JeOS template
. /usr/lib/oraclevm-template/functions

# ovm_configure_network finction is part of JeOS function library
# this function interactively collects user's input for the virtual
# machine network configuration: IP address, host name, gateway,
# netmask, DNS
ovm_configure_network

# Reconfigure agent
echo
echo "Reconfiguring Agent..."

# Parameters to be reconfigured
OMS_HOST=
OMS_PORT=
SECURE_PASSWD=

# Get parameters from user input
oms_connection

AGENT_HOME=/u01/app/oracle/product/agent10g

# replace OMS hostname, port and this hostname in emd.properties
sed -i "/^emdWalletSrcUrl/s|%OMS_HOST%|$OMS_HOST|" $AGENT_
HOME/sysman/config/emd.properties
sed -i "/^emdWalletSrcUrl/s|%OMS_PORT%|$OMS_PORT|" $AGENT_
HOME/sysman/config/emd.properties
sed -i "/^REPOSITORY_URL/s|%OMS_HOST%|$OMS_HOST|" $AGENT_
HOME/sysman/config/emd.properties
sed -i "/^REPOSITORY_URL/s|%OMS_PORT%|$OMS_PORT|" $AGENT_
```

```
HOME/sysman/config/emd.properties
sed -i "/^EMD_URL/s|%HOSTNAME%|$(hostname)|" $AGENT_
HOME/sysman/config/emd.properties
# Reconfigure agent done

# Allow ports open for Agent and ssh services.
system-config-securitylevel-tui --quiet \
  --port=3872 \
  --port=ssh

# reconfigure and rediscover targets
echo
su oracle -c "$AGENT_HOME/bin/agentca -f -d -t"

# secure agent
echo
echo "Securing Agent..."
su oracle -c "$AGENT_HOME/bin/emctl secure agent $SECURE_PASSWD"

# Start agent
echo
echo "Starting Agent..."
/etc/init.d/gcstartup start

# create  startup script at runlevel 3 5
ln -sf /etc/rc.d/init.d/gcstartup /etc/rc.d/rc3.d/S99gcstartup
ln -sf /etc/rc.d/init.d/gcstartup /etc/rc.d/rc5.d/S99gcstartup

# set env varible for user 'oracle'
cat >> /home/oracle/.bash_profile <<-EOF

# oracle env variables
AGENT_HOME=$AGENT_HOME
ORACLE_HOME=\$AGENT_HOME
JAVA_HOME=\$ORACLE_HOME/jdk
PATH=\$ORACLE_HOME/bin:\$ORACLE_HOME/OPatch:\$PATH
EM_SECURE_VERBOSE=1
export ORACLE_HOME AGENT_HOME JAVA_HOME EM_SECURE_VERBOSE PATH
alias cdo='cd \$ORACLE_HOME'
EOF

echo
echo "Reconfiguration of OEL5.2 and EM Agent 10.2.0.4 Comepleted."
press_anykey
```

B.4 Oracle Enterprise Manager Agent Clean Up Script

This example clean up script is used in the Oracle Enterprise Manager Agent template.

```
#!/bin/bash

AGENT_HOME=/u01/app/oracle/product/agent10g/

# parameters substitution
sed -i \
    -e '/^REPOSITORY_URL=/s|=.*|=http://%OMS_HOST%:%OMS_PORT%/em/upload|' \
    -e '/^emdWalletSrcUrl=/s|=.*|=http://%OMS_HOST%:%OMS_PORT%/em/wallets/emd|' \
    -e '/^EMD_URL/s|=.*|=https://%HOSTNAME%:3872/emd/main/|' \
    $AGENT_HOME/sysman/config/emd.properties
```

```
# remove log files
rm -f $AGENT_HOME/sysman/log/*

# remove old targets data
rm -f $AGENT_HOME/sysman/emd/upload/*
rm -rf $AGENT_HOME/sysman/emd/state/*
rm -f $AGENT_HOME/sysman/emd/lastupld.xml

# remove the runlevel startup script
rm -f /etc/rc.d/rc3.d/S99gcstartup
rm -f /etc/rc.d/rc5.d/S99gcstartup
```

B.5 Oracle VM Template Builder Configuration Script

This example configuration script is used to create the Oracle VM Template Builder guest virtual machine.

```
#!/bin/bash
#set -x

TB_VERSION="2.1.0"

. /usr/lib/oraclevm-template/functions
ovm_configure_network

ovm_info "Configuring Oracle VM Template Builder"

TB_HOME=/var/www/html/ovmtb
SCRIPTS_HOME=$TB_HOME/scripts

ovm_info "Initializing database."
cd $TB_HOME
python manage.py syncdb
#chmod a+w $TB_HOME/db/ovmtb.db

# start ovmtb daemon
#service ovmtb start

# init, change sites domain name
export PYTHONPATH=/var/www/html
export DJANGO_SETTINGS_MODULE=ovmtb.settings
cd $SCRIPTS_HOME
ovm_log "Changing site name"
python ovmtb-init.py -n $(hostname) -d $(hostname)

ovm_log "Restarting httpd"
service httpd restart
service ovmtb restart
# open httpd and sshd port for firewall
system-config-securitylevel-tui --quiet \
      --port=http \
      --port=ssh

echo
ovm_info "Oracle VM Template Builder $TB_VERSION configuration completed."
site=$(hostname)
# if get a invalid hostname, then give IP address.
if [ "$site" = "localhost.localdomain" ]; then
 site=$(ifconfig eth0|grep "inet addr" | sed 's/.*inet addr:\([^ ]*\).*/\1/')
fi
```

```
ovm_info  "Please go to http://$site/ovmtb"
press_anykey
```

C Troubleshooting

This Appendix contains troubleshooting information for Oracle VM Template Builder and contains:

- Security Settings
- Template Builds May Fail
- Resetting Oracle VM Template Builder
- Known Issues

C.1 Security Settings

The Oracle VM Template Builder guest virtual machine opens the httpd and sshd ports. These are required to be open to access the web-based user interface and to log in remotely to the command-line shell, respectively.

C.2 Template Builds May Fail

If you can access the Oracle VM Template Builder web interface, but cannot perform template builds, make sure both the Oracle VM Template Builder service, and the web server service are started.

The Oracle VM Template Builder service (ovmtb) starts the daemon which performs the template builds. The ovmtb service is separate to the web server (httpd) service, which serves up the Oracle VM Template Builder user interface. You must have both the ovmtb and httpd services running to build templates in Oracle VM Template Builder. See Section 2.8, "Starting and Stopping Oracle VM Template Builder" for information on how to start the services.

The template build log file is, by default, located in:

/var/log/ovmtb/

C.3 Resetting Oracle VM Template Builder

If you must reset Oracle VM Template Builder to the initial setup, use the ovmtb-reconfig.sh script. As the *root* user, enter

```
# /var/www/html/ovmtb/scripts/ovmtb-reconfig.sh
```

Oracle VM Template Builder is reset back to the initial installation configuration settings.

C.4 Known Issues

This section describes the known issues in Oracle VM Template Builder and any known workarounds. The known issues are:

- No Online Help In User Interface
- Admin User Does Not Receive Emails When Using DHCP
- Building Multiple Templates Simultaneously May Cause Unexpected Errors
- Template Build Progress Percentage Not Accurate
- Changing Configuration File May Cause Unexpected Errors
- Only One Additional Disk Is Included In a Template

C.4.1 No Online Help In User Interface

There is no online help provided in this release of Oracle VM Template Builder. Online help will be provided in future releases.

C.4.2 Admin User Does Not Receive Emails When Using DHCP

The Oracle VM Template Builder *admin* user's email address may fail to receive emails if you have installed Oracle VM Template Builder on to a guest virtual machine which uses DHCP to obtain an IP address.

Workaround: Modify the Oracle VM Template Builder email configuration settings and restart Oracle VM Template Builder and sendmail:

1. Modify the /var/www/html/ovmtb/setting.py file to contain the host name of the guest virtual machine on which Oracle VM Template Builder is installed. For example, if the email address is webmaster@example.com on host.example.com, enter:

   ```
   DEFAULT_FROM_EMAIL='webmaster@example.com'
   EMAIL_HOST='host.example.com'
   ```

2. Restart Oracle VM Template Builder and sendmail:

   ```
   # service ovmtb stop
   # service httpd stop
   # /etc/init.d/sendmail restart
   # service httpd start
   # service ovmtb start
   ```

C.4.3 Building Multiple Templates Simultaneously May Cause Unexpected Errors

The following error may be displayed when building a template:

```
...
Mounting VM image...
ERROR: losetup error - Please check the log (/var/log/modifyjeos.log) file!

ERROR creating JeOS image files...
```

This error may occur when multiple templates are being built at the same time and all available loop devices are in use. If all loop devices are in use, Oracle VM Template Builder is unable to mount the disk image files, so cannot complete the build process.

Workaround: Modify the number of loop devices on the system by adding the following line to the /etc/modprobe.conf file and restarting the operating system.

```
options loop max_loop=64
```

C.4.4 Template Build Progress Percentage Not Accurate

The template build progress percentage displayed in the **Template Projects** table on the **Template Projects** screen, and in the template build log pages does not accurately display the time remaining for a template build. Depending on the configuration of the template, certain steps in the build process may take some time to complete while others may complete quickly.

C.4.5 Changing Configuration File May Cause Unexpected Errors

Care should be taken when changing the parameters in the Oracle VM Template Builder configuration file, /var/www/html/ovmtb/settings.py. Changing the values in this file may result in failed builds, the inability to access completed builds from the web user interface, and *SuspiciousOperation* exceptions within the application. If more file system space is needed to hold the builds, it is recommended that you retain the default configuration parameters and, instead, add a new disk to the Oracle VM Template Builder guest virtual machine. To customize the guest virtual machine, see Section 2.9, "Customizing the Oracle VM Template Builder Guest Virtual Machine".

Future releases of the Oracle VM Template Builder will allow more fine grained control over the configuration settings.

C.4.6 Only One Additional Disk Is Included In a Template

Only one additional disk image is included in a template, even if multiple additional disks are requested to be included using the template build wizard.

Workaround: Edit the template configuration file (vm.cfg) to include the additional disks.

D

Oracle VM Template Builder Configuration

This Appendix contains information on the Oracle VM Template Builder configuration file. The configuration file is located in:

/var/www/html/ovmtb/settings.py

The settings listed in this Appendix are specific to Oracle VM Template Builder. All other settings in this file are standard to the Django framework. For more information on the standard Django settings in the configuration file, see the Django documentation at:

`http://docs.djangoproject.com/en/dev/ref/settings/#ref-settings`

D.1 Configuration File

The Oracle VM Template Builder configuration file options are listed below.

OVMTB_BUILD_QUEUE_DIR
The directory that contains the temporary files for the requested template builds. The Oracle VM Template Builder service scans this directory for any files, and performs the requested builds. The default for this parameter is:

/var/spool/ovmtb

This directory should be used only to host the build files, any other files might be interpreted by the Oracle VM Template Builder service as a build request. If the value of this parameter is changed, the directory must be writable by the user running the httpd service. By default this is the *apache* user.

OVMTB_BUILD_DIR
The directory used to perform the template builds. This directory should have sufficient space to hold all of the disk image files for the templates that are created. The default for this parameter is:

/opt/ovmtb/builds

OVMTB_TAR_DIR
The directory used to store the built templates in gzipped tar archives. This directory should have sufficient space to hold the tar files. The default for this parameter is:

/opt/ovmtb/tarballs

OVMTB_UPLOAD_DIR
The directory that holds the RPMs uploaded into template projects. The value of this parameter should be a subdirectory of the MEDIA_ROOT setting or exceptions are thrown when attempting to delete the packages. The directory should be writeable by

the user that is running the httpd service. By default this is the *apache* user. The default for this parameter is:

/opt/ovmtb/uploads

OVMTB_LOG_DIR
This is the directory that holds the log files. The default for this parameter is:

/var/log/ovmtb

OVMTB_MODIFYJEOS_DEBUG
Controls whether template build commands log debugging information. The default for this parameter is `False`.

OVMTB_MODIFYJEOS_DEBUG_DIR
The directory to which the debug information from the Oracle VM Template Builder build commands is written.

OVMTB_AUDIT
Controls whether requests to the Oracle VM Template Builder application are logged for later auditing. The default for this parameter is: `False`.

OVMTB_AUDIT_LOG
The file name in OVMTB_LOG_DIR to which requests to the Oracle VM Template Builder application are logged. Data is written each time a page is requested. The default for this parameter is `ovmtb_audit.log`.

OVMTB_SESSION_DATA_DEBUG
Controls whether session data is dumped to a log file for debugging purposes. The default for this parameter is `False`.

OVMTB_SESSION_DATA_DEBUG_LOG
The file name in OVMTB_LOG_DIR to which session data is dumped. Data is written each time a user requests a page. This is mostly used for debugging purposes. The default for this parameter is `session_data.log`.

Glossary

Domain

A configurable set of resources, including memory, virtual CPUs, network devices and disk devices, in which virtual machines run. A domain is granted virtual resources and can be started, stopped and restarted independently.

See also **dom0** and **domU**.

dom0

An abbreviation for *domain zero*. The management domain with privileged access to the hardware and device drivers. Dom0 is the first domain started by the Oracle VM Server at boot time. Dom0 has more privileges than domU. It can access the hardware directly and can manage the device drivers for other domains. It can also start new domains.

domU

An unprivileged domain with no direct access to the hardware or device drivers. Each domU is started by Oracle VM Server in dom0. The xm command-line tool is used to interact with each domU.

Guest

A guest operating system that runs within a domain in Oracle VM Server. A guest may be paravirtualized or hardware virtualized. Multiple guests can run on the same Oracle VM Server.

Hardware virtualized machine

A virtual machine with an unmodified guest operating system. It is not recompiled for the virtual environment. There may be substantial performance penalties running as a hardware virtualized guest. Enables Microsoft Windows™ operating system to be run, and legacy operating systems. Hardware virtualization is only available on Intel VT or AMD SVM CPUs.

Host computer

The physical computer on which Oracle VM Server is installed.

Hypervisor

The hypervisor, monitor, or Virtual Machine Manager (VMM). It is the only fully privileged entity in the system. It controls only the most basic resources of the system, including CPU and memory usage, privilege checks, and hardware interrupts.

JeOS

Oracle VM Template Builder uses *Just enough OS* or *JeOS* to facilitate building an operating system instance with only the absolute minimum packages you need for your template. This helps reduce the disk footprint by up to 2GB or more per guest virtual machine.

Management domain

See **dom0**.

Oracle VM Agent

An application installed with Oracle VM Server. It communicates with Oracle VM Manager for management of virtual machines. Oracle VM Manager manages the virtual machines running on Oracle VM Server by communicating with Oracle VM Agent. It contains three components: Server Pool Master, Utility Server, and Virtual Machine Server.

Oracle VM Server

A self-contained virtualization environment designed to provide a lightweight, secure, server-based platform for running virtual machines. Oracle VM Server is based upon an updated version of the Xen hypervisor technology. Includes Oracle VM Agent to enable communication with Oracle VM Manager.

Oracle VM Manager

Provides the user interface, which is a standard ADF (Application Development Framework) web application, to manage Oracle VM Server pools. Manages virtual machine lifecycle, including creating virtual machines from templates or from installation media, deleting, powering off, uploading, deployment and live migration of virtual machines. Manages resources including ISO files, templates and shared virtual disks. Also provides an API through a web service to Oracle VM Server.

Paravirtualized machine

A virtual machine with a kernel that is recompiled to be made aware of the virtual environment. Runs at near native speed, with memory, disk and network access optimized for maximum performance.

Preferred Server

A Virtual Machine Server that provides resources such as memory, CPU, network interface cards (NICs), and disk to the virtual machine. If you select only one Virtual Machine Server as the preferred server, the virtual machine always starts from and runs on this server. If you select multiple preferred servers, each time the virtual machine starts, it runs on the machine with the maximum available resources.

QEMU

Also referred to as qemu-dm, which is the process name. The virtualization process which allows full virtualization of a PC system within another PC system.

Server Pool

Logically an autonomous region that contains one or more physical Oracle VM Servers. Presents a unified view of the storage where the virtual machines reside, and groups the users of these virtual machines into a single community called a *group*, in which each user is a server pool member.

Server Pool Master

A component of Oracle VM Agent. An application that acts as the contact point to Oracle VM Manager, and to other Oracle VM Agents. Provides virtual machine host load-balancing, and local persistency for Oracle VM Server.

There is only one Server Pool Master in a server pool. A physical server can perform as the Server Pool Master, Utility Server and Virtual Machine Server simultaneously.

Utility Server

A component of Oracle VM Agent. An application that handles I/O intensive operations for virtual machines, server pools and servers, for example, copying, moving and renaming files.

There can be more than one Utility Server in a server pool. A physical server can perform as the Server Pool Master, Utility Server and Virtual Machine Server simultaneously.

vif

A virtual network interface for bridging network interfaces between domUs and dom0. When a domU is started it is assigned a number. This number is used to bridge the network interface from eth*n* to vif*n.0*.

Virtual disk

A file or set of files, usually on the host file system although it may also be a remote file system, that appears as a physical disk drive to the guest operating system.

Virtual Machine (VM)

A guest operating system and the associated application software that runs within Oracle VM Server. May be paravirtualized or hardware virtualized machines. Multiple virtual machines can run on the same Oracle VM Server.

Virtual Machine Manager (VMM)

See **Hypervisor**.

Virtual Machine Server

A component of Oracle VM Agent. An application which runs Oracle VM Server virtual machines. It can start and stop virtual machines, and collect performance data for the host and guest operating systems. Enables communication between the Server Pool Master, Utility Server and Virtual Machine Servers.

There can be more than one Virtual Machine Server in a server pool. A physical server can perform as the Server Pool Master, Utility Server and Virtual Machine Server simultaneously.

Virtual Machine Template

A template of a virtual machine. Contains basic configuration information such as the number of CPUs, memory size, hard disk size, and network interface card (NIC). Create virtual machine templates using Oracle VM Template Builder. Create virtual machines based on a virtual machine template using Oracle VM Manager.

VMM

See **Virtual Machine Manager (VMM)**.

Xen™

The Xen hypervisor is a small, lightweight, software virtual machine monitor, for x86-compatible computers. The Xen hypervisor securely executes multiple virtual machines on one physical system. Each virtual machine has its own guest operating system with almost native performance. The Xen hypervisor was created by researchers at Cambridge University, and derived from work done on the Linux kernel.

Index