

Oracle VM Benefits for ISVs

*An Oracle White Paper
April 2009*

INTRODUCTION

Oracle's virtualization technologies are an excellent delivery vehicle for Independent Software Vendors (ISV's) looking for a simple, easy-to-install and easy-to-support application delivery solution.

A typical ISV application can range from a simple single-server configuration to a large, distributed network of applications using shared storage and parallel processing to achieve extreme transaction processing (XTP). Any virtualization platform solution must be able to address this range of installations, and provide an upgrade path between them. Oracle VM is well suited for this challenge.

This white paper explains the hurdles that ISV's face when choosing an application delivery platform, and how Oracle VM and Oracle Enterprise Linux help them to overcome these issues.

THE PROBLEM WITH SUPPORTING MULTIPLE PLATFORMS

Why do ISV's support so many platforms? As a corollary, why would an ISV be interested in supporting Oracle Enterprise Linux?

If we look at the ISV application model, it has been essentially the same for decades.

- Design a software application that solves specific business problems.
- Analyze the market and see what operating systems and hardware the target market uses, or is forecast to use.
- Follow the 80/20 rule. Support the platforms and hardware that 80% of the customers use, and do one-off support for other operating systems and hardware in use by large or strategic customers.

This model worked well in the 1980's and early 1990's, when there were fewer operating systems and hardware platforms in use. Each hardware vendor had a *NIX that ran on its own proprietary hardware (e.g. Sun's Solaris, Hewlett Packard's HP-UX and IBM's AIX). Each of these commercial *NIX distributions came with enterprise-grade support. Although the distributions differed in their particulars, the programming model was fairly consistent (POSIX C) and it was possible to support most of the platforms with a modest effort by using good software development practices. For example, Oracle provided support for dozens of distributions of *NIX, as well as Microsoft operating systems.

Hardware and operating system support comes down to hard numbers: how many customers can I get by supporting a potential platform?

ISV's require that an operating system be properly supported by the vendor.

Cost is always an issue, and for an ISV reducing software royalties is critical.

The advent of the PC brought about massive changes, as Microsoft, Linux and commodity hardware manufacturers stormed the market with cheap alternatives to the commercial *NIX market. A PC running Linux or Windows could be had for a fraction of the cost of a commercial *NIX distribution on proprietary hardware.

Although similar to the commercial *NIX distributions, the open source Linux development model was vastly different, and the number of Linux distributions quickly outpaced the commercial varieties. Throughout this expansion, however, the commercial *NIX vendors had a distinct advantage over their Linux competition: enterprise level support. Most Linux distributions still do not provide production level, 24x7 customer support. This level of support is critical to the ISV market, as all support issues are first routed through the ISV's help desk. If issues related to the operating system arise, support for Linux could be limited to newsgroups and mailing lists – hardly an incentive for the ISV. But cost is always an issue, and the concept of a no-royalty operating system on commodity hardware remains very attractive.

It is not surprising then that a variety of Linux distributors are now providing better commercial support to back up their offerings. In fact, support is the main source of income for any commercial Linux offering. Yet even after a decade, these support organizations are still not at the level of the support that the commercial *NIX vendors provided. There are also time and real costs involved with multiple vendor support relationships. The lack of enterprise-level support is a primary reason for the creation of Oracle Enterprise Linux, which along with Xen is the foundation of the Oracle VM Server offering.

The Holy Grail: one operating system to test, certify and support.

Yet customers will run whatever operating system they choose, and will expect the ISV to support their choice. If the customer is running Windows XP SP2, they will expect the ISV to provide a Windows XP version of their product that is tested and certified with SP2. Of course it is very costly to create a native version of an application for all of the possible operating systems, versions and patch levels that customers use. This is the dilemma that every ISV must face. It goes without saying that the Holy Grail for an ISV is to support only one operating system and have every customer accept it.

Customers do not want to buy operating systems – they want to buy applications that solve business problems!

So is it possible for an ISV to reach this nirvana? Yes! If proper standards were in place, an ISV application could run just like a router on a network. When choosing a router, what does a customer look for? Features, as expressed as standards support. Need gigabit Ethernet over twisted pair cable? That would be 1000BASE-T support. How about VPN tunneling? That's 802.1q. Need identity services support? That would be 802.1x.

In the end, a customer will pick the router that meets the technical requirements and offers top-notch support. The customer does not know, and certainly does not care, what operating system the router is running. What is important is the functionality the application provides and the support organization that backs it up.

Some software companies have moved to this model, providing both the hardware and software needed to create a network appliance. Spam-filtering systems are a good example of this model. The vendor provides a 1U rack mount server that the customer puts in a rack and configures using a web-based user interface. The customer plugs the hardware into their network, enters some information into the web configuration tool, and voila: the application is filtering spam. The customer does not know or care what operating system is running on the network appliance, because it is irrelevant to their IT infrastructure. The application works within existing standards to receive and relay mail. When the application fails, the customer calls the vendor's support organization, which helps them with troubleshooting. The spam-filter vendor does not need to ask the customer what operating system they are running, and does not need to be concerned with how the operating system is interacting with the customer's network, as the customer does not have direct access to install or modify anything other than through the provided application interfaces. This helps both the ISV and the customer equally.

This model does not work for all ISV's, however. Not every ISV wants to be in the business of choosing hardware partners, installing hardware onto 1U server boxes and shipping large physical boxes from vast warehouses. Much of the appeal of the software industry is that you are selling bits that can be downloaded and installed with only minimal assistance from the vendor and with no physical distribution. Low overhead equals higher profits. So how can an ISV take advantage of the network appliance model without getting mired in hardware? One answer is virtual appliances.

TEMPLATES TO THE RESCUE

What Is Virtualization?

At its simplest level, virtualization is a technology that fakes out hardware, making an operating system think that it is running on a standardized hardware configuration. The software that handles this abstraction layer is called a *hypervisor*. An operating system runs inside of a *virtual machine*, which is just a container managed by the hypervisor. The hypervisor manages and distributes the processor, memory and I/O resources of the physical server between multiple running virtual machines.

Adding this layer of abstraction between the operating system and the hardware has several interesting practical uses:

- A hypervisor can run multiple virtual machines at once, making each virtual machine's operating system believe that it is running on its own hardware, when in reality they are all sharing the processor(s), memory and I/O of a one or more physical servers. An ISV can now separate their product into separate applications, each running in its own virtual machine. If more capacity is needed, one of the virtual machines can be

Virtualization helps solve the hardware support problem by providing a standardized hardware environment for operating systems.

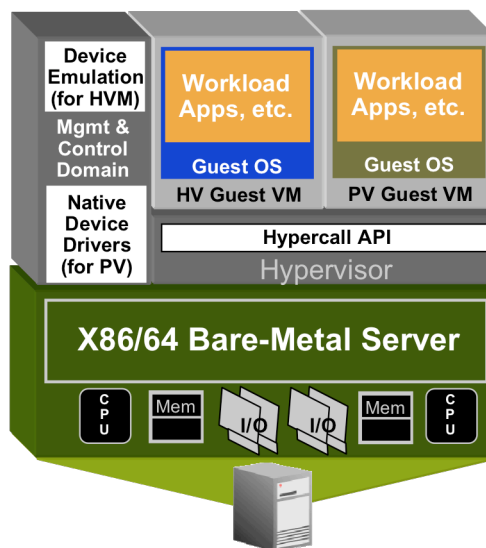
A hypervisor abstracts physical hardware from the operating system and manages virtual machines.

moved to a server with higher capabilities or less load to improve performance.

- Hypervisors on different physical servers can work in conjunction to expand the capacity of the application, provide redundancy and increase uptime. A virtual machine does not know which physical server it is running on, and it is possible to move a running virtual machine from one physical server to another physical server without stopping the running applications. The ISV can now scale the customer's application by transferring a virtual machine to run on a more powerful server, without requiring a re-installation of the hardware, or even a moment of downtime!
- The hypervisor always presents a consistent device interface for hard drives, network cards and processors, so the ISV no longer needs to certify particular hardware configurations – that becomes the job of the hypervisor vendor.

Oracle's virtualization technology, Oracle VM, is based on the open source Xen project.

Oracle VM is based on the open source Xen hypervisor. A diagram of a Xen-based system is given below. The Xen hypervisor runs on top of a bare-metal x86 or x86_64 system. The hypervisor is only responsible for allocating memory and CPU to the various virtual machines (called domains). Xen immediately boots a minimal domain, referred to as Domain 0, which in turn manages the I/O and peripherals, and distributes network and disk resources to the user domains.



For more information, please refer to several excellent [white papers on virtualization technology](#)¹ available through Oracle's web site.

¹ Please see <http://www.oracle.com/technology/tech/virtualization/index.html> for white papers on virtualization technology.

Templates relieve the customer from requiring technical knowledge external to the application, such as modification of operating system parameters or installation of required supporting software.

What is a Template?

A template is a distributable file or files that encapsulates a fully installed, patched, configured and performance-tuned application stack, from the operating system to the end user application. This includes all supporting software necessary to make the system operate. Best practices for installation, configuration and performance tuning let the vendor put their best foot forward every time.

Templates can be composed of multiple virtual machines, for example two processing nodes and an NFS server providing shared storage. The idea is to encapsulate a full system into a redistributable package that can be installed and up and running in minutes instead of days. Think of it as a system-in-a-box, where the box is a compressed file available for download from your company's web server.

A template works within a virtualization framework that provides management and HA capabilities that work with the existing software-based HA capabilities of the applications (such as WebLogic Server clustering), while providing hardware-based HA that is completely transparent to the virtual machines.

To use a template, the compressed file is simply copied onto the virtualization server, extracted and started using the management interfaces. Minimal information such as network configuration is collected on the initial boot. Additional configuration is then done through a web-based interface, enabling the template to run from a headless virtualization server in a data center.

For example, Oracle has downloadable templates that encapsulate Oracle Database 11g Enterprise Edition running with Automated Storage Management (ASM). This is normally a lengthy, involved process, requiring a good deal of database knowledge. By creating a template, the installer only needs to know the network information and the name of the database. No knowledge of ASM or Oracle TNS is necessary. An ISV can similarly take their applications and create a "golden installation" of their software stack. This has several advantages:

- Reduces support calls for installation-related issues.
- Encapsulates best practices into an executable form.
- Encourages new customers to try the software with no risk and fewer installation headaches.
- Gives consistent results for Proof-Of-Concepts or comparison testing, whether the vendor is involved or not.

How Are Templates Created?

So how can an ISV create a template to encapsulate their application?

The ISV needs to create a pre-installed, fully functional version of their software application on an operating system. This is called a *template* of the software application. A *golden image* is a template of the ideal installation, done according to best practices and tuned for the best possible performance. The golden image should be kept up-to-date, including all applicable patches.

Templates encode best practices into a distributable virtual machine, reducing support costs and improving customer satisfaction.

There are three ways to create such a template golden image:

Easily convert an existing installation of your products using physical to virtual converters (p2v).

- Create an installation of the operating system, application(s) and dependent software on physical hardware, apply patches, configure the software, tune the system, then convert the physical system to a virtual machine (this is called a *p2v*, or physical-to-virtual, conversion). This is the easiest method for a beginner, but results in the largest downloadable image size and therefore requires substantial tuning and modification of the operating system configuration to slim down the image and secure the system. Oracle has published an excellent [white paper on p2v](#)².
- Create a virtual machine and install the operating system, application(s), and dependent software, apply patches, configure the software, and tune the system. This method is easy once the basics of the hypervisor are mastered, as it is very simple to create a stock operating system image and start working on the installation. The resulting image requires no conversion, as it is already in the correct format for distribution. All of the tuning and operating system modifications from a p2v installation are necessary for this method as well.
- Use a custom-made, pared-down operating system template made specifically for virtual environments, then install the application, apply patches, configure the software, and tune the system. This approach results in the smallest downloadable size, and provides the most resilience as only the absolute minimum amount of software and services are running on the machine. Oracle provides templates that are based on a minimalist operating system, pre-tuned to work well in a virtual environment. Oracle also provides scripts to help with the task of configuring the operating system and network for a particular operating system. This set of tools is referred to as JeOS, which stands for “Just Enough Operating System”. Information on [making your own templates using JeOS](#)³ can be found on the Oracle Technology Network (OTN).

JeOS (pronounced “juice”) is an excellent option for creating your own templates using a fully tested and supported version of Oracle Enterprise Linux.

Oracle creates and maintains templates of many of its products, including:

- Oracle Application Server 10g Release 3 Webcenter
- Oracle Database 10g Release 2 (10.2.0.4)
- Oracle Database 11g (11.1.0.6)
- Oracle Enterprise Linux (several versions)
- Oracle Enterprise Manager Grid Control
- Oracle WebLogic Server 10g Release 3
- Oracle VM Manager 2.1.2

² Please see <http://www.oracle.com/technologies/virtualization/docs/p2v-whitepaper.pdf> for a white paper on physical to virtual (p2v) conversions.

³ Please see http://www.oracle.com/technology/software/products/virtualization/vm_jeos.html for a white paper on creating Oracle VM templates.

- Oracle Fusion Middleware Service Oriented Architecture on Oracle Containers for JEE 10.1.3.4
- Oracle Fusion Middleware Service Oriented Architecture on WebLogic Server 9.2 (10.1.3.3)

You can find a [list of all available templates](#)⁴ on OTN.

From the Customer's Point of View

What does an installation of a template look like to a customer?

The customer starts by installing the virtualization software onto a server. This includes installation of the hypervisor and associated management applications. A simple Oracle VM installation can be done in less than ten minutes (see the [Oracle VM Wiki](#)⁵ for a video of a typical Oracle VM installation). More complicated installations, for example those involving multiple physical servers, multiple network cards per server, and a SAN with shared storage and high availability (HA) services take more time to prepare and execute.

The customer then downloads (or installs from media) the template files from the ISV onto the server and starts the virtual machine using the virtualization management console. The first time the virtual machine is launched, the administrator will need to enter some information, such as network address and hostname, plus any information needed for the installation of the application. Actual configuration of the application can be done at a later point using a web browser pointing at the ISV application's web interface, or through remote configuration tools provided by the ISV.

At that point the ISV application is installed and running, and the customer can properly evaluate the software without worrying about whether their hardware is supported, whether the operating system needs is compatible with their corporate mandates, or whether they have properly tuned the software before beginning testing.

WHY USE ORACLE ENTERPRISE LINUX AND ORACLE VM?

Take a look at the available virtual appliances on VMware's Marketplace, and you will find a preponderance of VM's from open source contributors, generally running on CentOS or Ubuntu. Microsoft Windows and Red Hat Enterprise Linux cannot be used because the licensing terms forbid distribution without purchase of a license or support contract. Oracle Enterprise Linux has no such restrictions.

⁴ Please see <http://www.oracle.com/technology/products/vm/templates.html> for a current list of available templates produced by Oracle.

⁵ Please see <http://wiki.oracle.com/page/Oracle++VM+installation> for a video showing a typical Oracle VM installation.

Installation of OracleVM and starting a template takes minutes, not days.

Gartner rates Oracle Enterprise Linux tops in the categories that matter to enterprise customers.

The Difference is in the Support

So how about the quality of Oracle Enterprise Linux? Gartner has studied the top three Enterprise Linux vendors, Oracle, Novell and Red Hat⁶. Their conclusion is that Oracle comes out on top in the categories that matter most for enterprise customers: Technical Support and Quality, Patent Infringement Protection, Subscription Pricing and Flexibility, and Reliability, Availability and Scalability (RAS). This is true even though Oracle Enterprise Linux is 100% binary compatible with Red Hat Enterprise Linux⁷. They are built from the same source code, and are virtually indistinguishable at the binary level. This implies that Gartner considers the differences between Red Hat and Oracle's support programs to be the determining factor. Yet despite the better support, Oracle Enterprise Linux support is *less* expensive than Red Hat Linux!

Production Ready Virtual Machines

Looking again at the VMware Marketplace, you will notice that there are very few production quality virtual machines available. In fact, the "Certified Production Ready" category now points to a page indicating that the new name of the program is "VMware Ready Virtual Appliance Program". A subtle change, but noteworthy. The phrase "production ready" implies that the virtual machine can be placed on the customer's virtual infrastructure and be rolled directly into production, with little or no modification (although this is exactly what Oracle VM Templates do). The phrase "VMware Ready" does not really say anything more than the appliance is certified to run on the VMware platform. There is no indication of the performance or quality of the vendor's application when run in production.

Oracle VM Templates provide production-ready virtual machines, not demos and cripple-ware.

Most of the virtual appliances on the VMware Marketplace are demos or cripple-ware, intended only for review by the downloader. They are sized for minimal download time and minimum footprint on the virtual server. While this is convenient, it does not allow for the kind of rigorous testing that is generally required before the purchase of enterprise software.

Oracle VM is a Xen-based hypervisor that runs at nearly bare-metal speeds. Independent testing has shown Oracle VM to run applications at 94% of native (non-virtualized) speeds⁸. This means that an ISV can truly create a production ready virtual machine and have confidence that their product will always make an outstanding first impression.

⁶ Please see <http://mediaproducts.gartner.com/reprints/oracle/article64/article64.html> for the original Gartner report.

⁷ Please see <http://www.oracle.com/technologies/linux/el5cert-ds.pdf> and <http://www.oracle.com/technologies/linux/el4cert-ds.pdf> for more details.

⁸ Please see http://www.oracle.com/us/corporate/press/017502_EN and www.oracle.com/technologies/virtualization/docs/ovmbenchmark.pdf for performance benchmarks of Oracle VM.

**Freely download, develop, and distribute
Oracle Enterprise Linux and Oracle VM.**

Non-Restrictive Licensing Terms

Both Oracle Enterprise Linux and Oracle VM can be freely downloaded, embedded, and distributed without breaking the terms of the licensing agreement. This does not refer to crippled or demo versions of the software. This includes the full Linux distribution, Oracle VM Server and Oracle VM Manager. Oracle is the only major virtualization vendor that has a no-cost, non-crippled management console for virtualization.

Oracle does not charge for Oracle Enterprise Linux or Oracle VM, and support contracts are optional. This means that Oracle is unique in the market, as it provides the only enterprise-grade Linux distribution and virtualization platform that can be used to create virtual machines that are not encumbered by restrictive IP licensing terms.

Summary

Oracle VM provides ISVs with a real solution to the headaches introduced by multiple platforms support and multi-vendor product support.

ISV applications running inside of Oracle VM provide savings in development, testing and support for the partner. Oracle provides a fast, supported, best of breed environment for the partner applications with Oracle Enterprise Linux and Oracle VM.

For more information on how you can get started with Oracle VM technology, contact your Oracle account manager, or send an email to oelhelp_ww@oracle.com for Oracle Enterprise Linux questions, or ovmhelp_ww@oracle.com for Oracle VM questions.



Oracle VM Benefits for ISVs
March 2009
Author: John Margaglione

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2009, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

Other names may be trademarks of their respective owners.