Oracle VM for ISV Applications

ORACLE WHITE PAPER | JANUARY 2015

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

INTRODUCTION

Oracle's virtualization technologies can be 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 particularly well suited for this challenge.

This white paper explains the hurdles that application vendors face when choosing an application delivery platform, and how Oracle VM and Oracle Linux can help them overcome these issues.

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

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 an additional platform like Oracle Linux[1]?

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

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

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

This model worked well in the 1980's and early 1990's, when there were not many operating systems or 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

[1] Please see the Oracle site for partners on Certifying your Hardware and Applications on Oracle Linux for more details.

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 has long had support for dozens of distributions of *NIX, as well as Microsoft operating systems.

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

The advent of the PC brought about massive changes, as Microsoft, Linux and 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 did 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 were to arise, support for Linux was 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 was and remains very attractive.

It is not surprising then that a variety of Linux distributors began 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 provide. Enterprise-level Linux support needs to be able to apply security updates, diagnostics patches, critical bug fixes and dynamic tracing for administrators and developers without rebooting. There are also time and real costs involved with multiple vendor support relationships. This lack of enterprise-level support was a primary reason for the creation of Oracle 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 Microsoft Windows Server 2012 R2, they will expect the ISV to provide a Windows Server 2012 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? Sure, there are technical feature sets, but those are 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 troubleshoot. The one question the spam-filter vendor will never ask the customer: what operating system are you running? 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.

Virtualization helps solve the hardware support problem by standardizing the hardware.

VIRTUALIZATION TEMPLATES (APPLIANCES) 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 the running virtual machines.

A hypervisor abstracts physical hardware from the operating system and manages 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 single machine. 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 moved to a separate server to improve performance.
- Hypervisors on different physicals 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, the ISV no longer needs to certify particular hardware configurations – that becomes the job of the hypervisor vendor.

There are several excellent white papers on virtualization technology.

Oracle VM 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 an Oracle VM Template?

A virtualization template appliance is an application or application stack that has been encoded into one or more virtual machines. The idea behind a virtualization template is that the software has been pre-installed and pre-configured on an operating system according to the software vendor's best practices. An Oracle VM Template includes the operating system, the correct patch for the OS and can include the database, application server, and application. The Oracle VM Template is simply copied onto the virtualization server environment and started as a virtual machine. Minimal information such as network configuration is collected on the initial boot. Additional configuration is then done through a web-based interface, enabling the virtual machine to run from a headless virtualization server in a data center.

Oracle VM Templates encode best practices into an executable virtual machine, reducing support costs and improving customer satisfaction.

For example, Oracle has downloadable Oracle VM Templates that encapsulate Oracle Database Enterprise Edition running with Automatic Storage Management (ASM). This is normally a lengthy, involved process, requiring a good deal of database knowledge. By providing an Oracle VM Templates, the installer then 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.

How Are Oracle VM Templates Created?

So how can an ISV create a virtualization 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.

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

There are several straight forward ways to create such a template golden image:

- Create an installation of the operating system and applications 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. The Oracle VM Admin Guide covers p2v and there are several blogs with excellent tips and video examples on social media running through a p2v conversion.
- Create a virtual machine and install the operating system and applications, 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.

Using a provided operating system template is an excellent option for creating your own templates using a fully tested and supported version of Oracle Linux.

- Use a provided operating system template made specifically for virtual environments, then install the application, apply patches, configure the software,

tune the system, and reset the first time boot scripts parameter. 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. Scripts that configure the operating system and optionally applications at first boot can be utilized. Oracle provides Linux templates that are based on a minimalist operating system which are pre-tuned to work well in a virtual environment. Oracle also provides Virtual Machine Guest Additions that can do bidirectional communication to a guest virtual machine which can help with the task of configuring the operating system, network, and applications.

Oracle provides Linux templates that are based on a minimalist operating system which are pre-tuned to work well in a virtual environment.

The Oracle VM documentations include information on making your own templates.

Oracle creates and maintains golden images of several products (which Oracle calls Oracle VM Templates), including:

- Oracle Application Server 10g Release 3 WebCenter
- Oracle Fusion Middleware Service Oriented Architecture (SOA) 10.1.3.4
- Oracle Business Intelligence Enterprise Edition 10.1.3.4
- Oracle Database 11g & 12c Single Instance and Real Application Clusters (RAC)
- MySQL 5.5, Enterprise Edition
- Oracle Linux 4, 5, 6
- Oracle Solaris 10, 11
- Oracle Secure Global Desktop
- Oracle Enterprise Manager 12c
- JD Edwards EnterpriseOne and EnterpriseOne Tools
- PeopleSoft PeopleTools, ELM, FSCM, Portal Solutions
- E-Business Suite

You can find a list of available templates on OTN with additional templates and information from the relevant product section on the My Oracle Support site.

From the Customer's Point of View

What does an installation of a software appliance 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 Hands on Labs for an example 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

To simplify the server, storage, and networking installation along with the configuration and maintenance for ISV applications Oracle provides a Virtual Compute Appliance (VCA).

and execute. Oracle works closely with server and storage vendors to verify their solutions and publish the tested systems on a Hardware Compatibility List (HCL).

Oracle engineered systems are purpose-built systems based on Oracle hardware and software components that are fully integrated and optimized. Because they are pre-configured to meet a particular purpose, they lower the cost and complexity of implementing IT infrastructure, speeding time-to-production for both internal and external Oracle customers. Oracle VM is a core component in several engineered systems because it brings flexibility, performance, and reliability to virtualized deployments of Oracle applications. ISV applications running on supported guest OS of Oracle VM can be quickly deployed into Oracle engineered systems. For example, to simplify the server, storage, and networking installation along with the configuration and maintenance for ISV applications Oracle provides a Virtual Compute Appliance (VCA). The VCA is a converged infrastructure solution that combines preconfigured networking, servers, and storage that system administrators can setup quickly. VCA is a general purpose appliance for ISV applications with Oracle products and hardware pre-integrated including Oracle VM Templates. It helps ISV applications get deployed quickly with a minimum number of vendors involved and meet software compliance requirements. VCA with Oracle VM Templates is Cloud , IaaS, PaaS, and SaaS ready with features like metering and self service portal management capabilities.

To provide ISV applications and administrators more automation and interoperability directly from their applications, Oracle VM Manager exposes a fully supported Web Services API that offers both SOAP and REST interfaces to program any action supported within Oracle VM. In addition, Oracle VM Manager Command Line Interface (CLI) can be used to perform the same functions as the Oracle VM Manager Web Interface.

To provide ISV applications and administrators more automation and interoperability directly from their applications, Oracle VM Manager exposes a fully supported Web Services API that offers both SOAP and REST interfaces to program any action supported within Oracle VM. In addition, Oracle VM Manager Command Line Interface (CLI) can be used to perform the same functions as the Oracle VM Manager Web Interface. The CLI commands can be scripted and run securely in conjunction with the Web Interface, thus bringing more flexibility to help you deploy and manage an Oracle VM environment. ISV partners can leverage WS-API and/or CLI to integrate their solutions with Oracle VM. You can learn additional details from Oracle VM Documentation.

Once the server is setup the customer then downloads (or installs from media) the virtualization template appliance files from the ISV onto the server and starts the virtual machine. 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 first boot or a later point using a web browser pointing at the virtual appliance's web application.

At that point the software is installed and running, and the customer can properly evaluate the software without worrying about whether their hardware is supported, how the

operating system needs to be integrated with their other systems, or whether they have properly tuned the software before beginning testing.

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 single, supported, best of breed environment for the partner applications with Oracle Linux and Oracle VM.

For more information on how you can get started with Oracle VM technology, contact your Oracle salesperson or send email to ol-ovm-info@oracle.com.

Author: John Margaglione, Dave Fowler

ORACLE

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/oracle
twitter.com/oracle
oracle.com

Hardware and Software, Engineered to Work Together