


Server Templates mit Oracle VM



Wer Images für virtuelle Server erstellen will, braucht gemeinhin weitreichende Systemkenntnisse und Zeit. Der Prozess ist fehlerträchtig und schwer zu standardisieren. Mit den Server Templates für die freie Oracle VM lassen sich individuell angepasste Images für virtuelle Maschinen allerdings mit wenigen Klicks in einer Web-Oberfläche installieren und starten. Ulrich Wolf

Oracle VM ist einzige Server-Virtualisierungslösung, die für Oracles kompletten Solution Stack zertifiziert ist. Deshalb spielt sie für alle Enterprise-Anwendungen und die Virtualisierungs-, Grid- und Cloud-Strategien des Konzerns eine fundamentale Rolle. Jedoch ist Oracle VM als Basis einer Virtualisierungsumgebung auch dann eine praktische Alternative zu purem Xen, wenn überhaupt keine Oracle-Produkte zum Einsatz kommen.

OES und Xen als solide Grundlage

Die technische Basis des Oracle VM Server bilden Oracle Enterprise Linux (OEL), das binärkompatibel mit Red Hat Enterprise Linux ist, und der Open-Source-Hypervisor Xen. Oracle VM ist freie Software, lizenziert unter der GPL. Sowohl die Quellen als auch Installationsmedien stehen als ISO-Dateien auf den E-Delivery-Servern von Oracle zum Download zur Verfügung (1). Als Gast-Betriebssysteme sind OEL in den Versionen 4 und 5 beziehungsweise Red Hat Enterprise Linux (RHEL) in den Versionen 3 bis 5 sowie diverse Windows-Varianten von Windows 2000 über XP bis Vista und Windows 2008 SP1 zertifiziert. Auch Solaris 10 läuft als Gast-System, der Abschluss der Zertifizierung steht kurz bevor.

Oracle VM weist gegenüber einer reinen Xen-Installation zahlreiche nützliche Erweiterungen auf: Externer Shared Storage auf der Grundlage von SAN, iSCSI oder NFS lässt sich in Form eines gemeinsamen Storage Repository auf einfache Weise einbinden. Zudem ist es sehr leicht, Hochverfügbarkeits-Szenarien umzusetzen, also virtuelle Maschinen beim Ausfall eines Servers automatisch auf einem anderen zu starten. Auch Load Balancing macht mit Oracle VM keine großen Schwierigkeiten..

Zudem bringt Oracle VM weitere Werkzeuge mit: Für die Konvertierung nicht-virtualisierter in virtualisierte Systeme (Physical-to-Virtual, P2V), die Konvertierung von VMware-Maschinen und die sichere Migration virtueller Server von einer physischen Maschine zur anderen. Der VM Server selbst, die Vorgehensweise beim Aufsetzen der verschiedenen HA- und Storage-Optionen sowie alle Tools sind ausführlich dokumentiert. Auch diese Dokumentationen sind frei zugänglich (2).

Seitdem Ende 2007 die erste Version von Oracle VM erschien, hat sich Oracles Virtualisierungs-

portfolio ständig erweitert, schwerpunktmäßig um Lösungen für Konfiguration, Management und Deployment von virtuellen Maschinen. All diese Aufgaben können Oracle VM Templates enorm erleichtern.

Templates sind Vorlagen für Images, die dafür sorgen, dass eine oder mehrere virtuelle Maschinen vorkonfigurierte, aufeinander abgestimmte OEL-Pakete oder Third Party Software erhalten. Oracle bietet zahlreiche fertige Templates für Enterprise-Software an, beispielsweise für Anwendungen wie Siebel CRM, PeopleSoft HCM, Middleware wie den WebLogic Server 10g und natürlich auch die Oracle Database 11g und 10g. Jedoch ist es auch sehr leicht, eigene Templates zu erstellen, um sie selbst zu nutzen oder als Serviceprovider Kunden zur Verfügung zu stellen (3).

Schlanke Basis JeOS

Grundlage für alle Templates, selbst gebaute oder von Oracle angebotene, ist OEL JeOS (4), was für „Just enough Operating System“ steht. Dahinter verbergen sich schlanke, virtualisierungsfreundliche Minimalversionen von Oracle Enterprise Linux in verschiedenen Versionen (4.7 bis 5.3) mit Kernen, die wahlweise für hardwareunterstützte Virtualisierung oder Paravirtualisierung konfiguriert sind.

Erstellt man ein Template, lassen sich zu diesen Basissystemen beliebige Softwarepakete hinzufügen, die reguläre Bestandteile von OEL sind. Aber auch selbst erstellte oder in Repositories von Drittanbietern vorhandene Pakete lassen sich einbinden.

Um die Konfiguration des Systems und der Software individuell anzupassen, ist es möglich, beim Aufsetzen oder Bearbeiten des Templates Konfigurationsskripte anzugeben. Beispielsweise lassen sich so Nutzer hinzufügen, Datenbank-Schemata in RDBMs erstellen, Webserver konfigurieren oder die Konfigurationsparameter für jeden beliebigen Service setzen.

Das Schreiben dieser Skripte erfordert natürlich Kenntnisse von OEL und der verwendeten Applikationen.

Wenn sie jedoch einmal vorhanden sind, kann sie jeder Benutzer des Template Builder mittels einer einfachen Web-Oberfläche in neue Templates einbinden, exakte und verständliche Dokumentation vorausgesetzt.

Definierte Deployment-Prozesse

Die Vorteile dieser Vorgehensweise liegen auf der Hand: Mit JeOS ist eine einheitliche, standardisierte und zertifizierte Basis vorhanden, die Wildwuchs von vornherein eindämmt. Aus der Sicherheitsperspektive verringert das schlanke Grund-Betriebssystem (nur um die 250 MByte) die Angriffsfläche.

Das einheitliche Vorgehen bei der Konfiguration des Betriebssystems oder von Anwendungen erlaubt es, im Unternehmen eine robuste Best-Practice-Wissensbasis für die System- und Anwendungsconfiguration aufzubauen und im Zuge des „Separation of Concerns“ die Verantwortlichkeiten entsprechend der Projekt- oder Unternehmensorganisation zu verteilen. Für das eigentliche Anfertigen und Verteilen der Images sind dann kaum noch spezielle Kenntnisse nötig. Dies erfolgt im Idealfall komplett durch Zusammenklicken auf der Weboberfläche. Somit stellen Templates einen robusten und schnellen Weg zum Deployment aller Arten von Server-Anwendungen dar.

Da sich die erforderliche Software schnell und unkompliziert installieren lässt und die Hardware-Anforderungen moderat sind, lohnt sich durchaus eine Installation auf einem Testsystem, um die Software und die prinzipielle Vorgehensweise im Zusammenhang mit den Templates kennenzulernen.

Die folgenden Abschnitte skizzieren grob den Weg zu einem solchen Testsystem, mit dem der Anwender VM-Templates einrichten und in virtuellen Maschinen starten kann. Weitere Einzelheiten sind in der Dokumentation der jeweiligen Softwarepakete zu finden.

VM Server, VM Manager und Template Builder

Erforderlich sind softwareseitig der Oracle VM Server in Version 2.2, der Oracle VM Manager für VM 2.2 und der Oracle VM Template Builder für VM 2.2, die alle auf Oracles E-Delivers-Website (1) zur Verfügung stehen. Hardwareseitig sollte eine Maschine mit 64-Bit-Architektur zur Hand sein, die hardwareunterstützte Virtualisierung mittels Intels VT-x- oder AMD-V-Technologie erlaubt. Für bequemes Arbeiten sollten 4 GByte RAM verbaut und mindestens 80 GByte unpartitionierter Plattenplatz auf einer lokalen Hard Disk vorhanden sein. Außerdem ist ein Client-PC mit installiertem VNC-Viewer notwendig.

Der VM Manager ist eine Client-Server-Anwendung zum Management eines oder mehrerer Pools von VM-Servern und vereinfacht unter anderem auch das Deployment der Template Images. Zur Bedienung ist eine intuitive, Java-basierte Weboberfläche vorhanden, zusätzlich stehen aber auch Kommandozeilenwerkzeuge zur Verfügung (5).

Es gibt mehrere Wege, den VM Manager zu installieren. Die schnellste, einfachste und von Oracle empfohlene Vorgehensweise ist die Installation auf einer separaten physikalischen Maschine mit bereits vorhandenem OEL oder RHEL über die Installations-CD. Dieser Weg ist im Oracle VM Manager Installation Guide detailliert beschrieben.

VM Manager als Server Template

Andererseits ist VM Manager auch als Server Template für den Betrieb in einer virtuellen Maschine installierbar. Es ist ohne weiteres möglich, und insbesondere für Test- Entwicklungs- oder

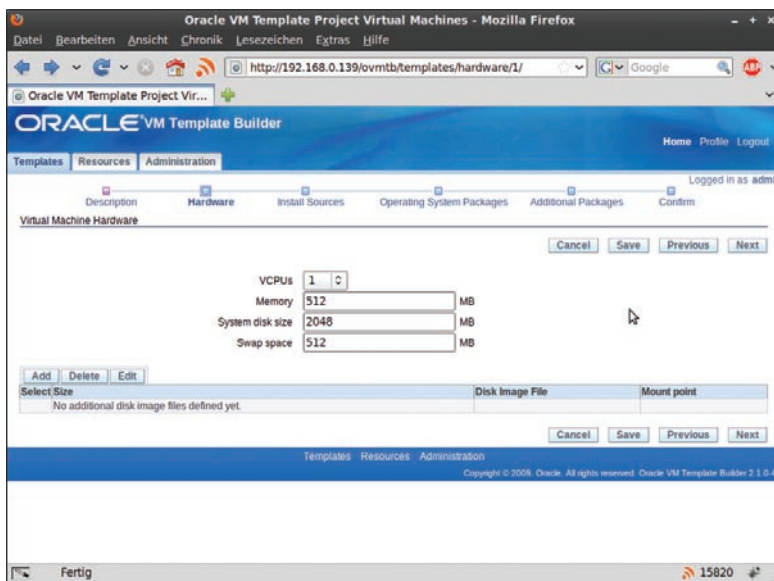


Abbildung 1: Festlegen der virtuellen Hardwareausstattung des neuen Servers in einer einfach zu bedienenden Weboberfläche.

produktive HA-Umgebungen auch sinnvoll, den VM-Server in einer VM auf einem – oder dem einzigen – physikalischen Server unterzubringen, den er managen soll.

Dazu ist auf dieser Maschine zuerst der VM Server zu installieren. Das Installationsmedium ist eine einzige CD, die Installation selbst dauert inklusive Grundkonfiguration nicht länger als fünf bis zehn Minuten, falls für eine Testinstallation vorerst noch kein externer Storage einzubinden ist.

Soll der VM Manager auf derselben physikalischen Maschine laufen, muss er dort als Template in einer virtuellen Maschine installiert sein. Das Template steht nur für eine (virtuelle) 32-Bit-Plattform zur Verfügung, unabhängig von der die physikalische Hardware, die selbstverständlich eine 64-Bit-Architektur haben kann.

Die Readme-Datei, die auf der Download-Seite des Templates verlinkt ist, enthält eine Schritt-für-Schritt-Anleitung für diese Installationsvariante. Sie läuft nach folgendem Schema ab: Im ersten Schritt legt ein Shellskript die virtuelle Maschine an, installiert das Image, konfiguriert das System und startet den virtuellen Server. Anschließend ist es erforderlich, sich über VNC am Server anzumelden und die Konfiguration zu vervollständigen. Erst danach erfolgt die erste Anmeldung mit dem Web-Browser über HTTP oder HTTPS.

Ist alles gut gegangen, sollte der vorher installierte VM Server unter »Servers« jeweils als Server Pool Master, Utility Server und Virtual Machine Server eingetragen sein. Diese drei Einträge rühren daher, dass der VM Manager gleich ganze Pools von VM Servern verwalten kann, in denen dann einzelne Server bestimmte Verwaltungsaufgaben haben. Im vorliegenden Fall übernimmt aber der einzige VM Server alle Aufgaben.

Der VM Manager speichert die Konfigurationsdaten der von ihm verwalteten virtuellen Server in einer Oracle Express Edition (XE) 10g oder in einer anderen Oracle-Datenbank-Variante. Oracle XE ist im Template enthalten, das Installationsskript richtet die Datenbank automatisch ein. Oracle XE ist zwar nicht Open Source, aber lizenzkostenfrei. Die Lizenz erlaubt – unter Beachtung der entsprechenden Bestimmungen hinsichtlich Trademarks und Copyright-Hinweisen – auch die Weitergabe an Kunden und das Einbinden in eigene Produkte.

Template Builder als Server Template

Ist der VM Manager installiert, lässt sich mit seiner Hilfe das Template für den Template Builder in einer neuen virtuellen Maschine installieren. Er findet sich in den Templates für die 64-Bit-Architektur. Die Vorgehensweise ist auch hier wieder im Readme des Templates beschrieben. Das für den Template Builder erstellte Image ist mit 60 GByte recht umfangreich, weil es einen großzügig dimensionierten Datenbereich für die Images enthält, die der Template Builder später erstellt. Wer sich das Einrichten einer weiteren virtuellen Maschine sparen will, für den stellt das Template-Builders-RPM eine Alternative dar, die die Software auf ein bereits existierendes OEL-System bringt. Wie beim VM Manager erfolgt auch beim Template Builder der Zugriff über eine Web-Oberfläche. Beim ersten Aufruf ist ein Nutzer anzulegen, anschließend ist die Software bereit, mittels »Templates|New Template Project« ein erstes Template zu erstellen. Dabei führt die Oberfläche des Template Builders den Anwender in Wizard-Manier durch die erforderlichen Schritte.

Wizard-geführte Template-Erstellung

Zur Auswahl stehen zwölf verschiedene JeOS-Images: OEL in den Versionen 4.7, 5.2 und 5.3

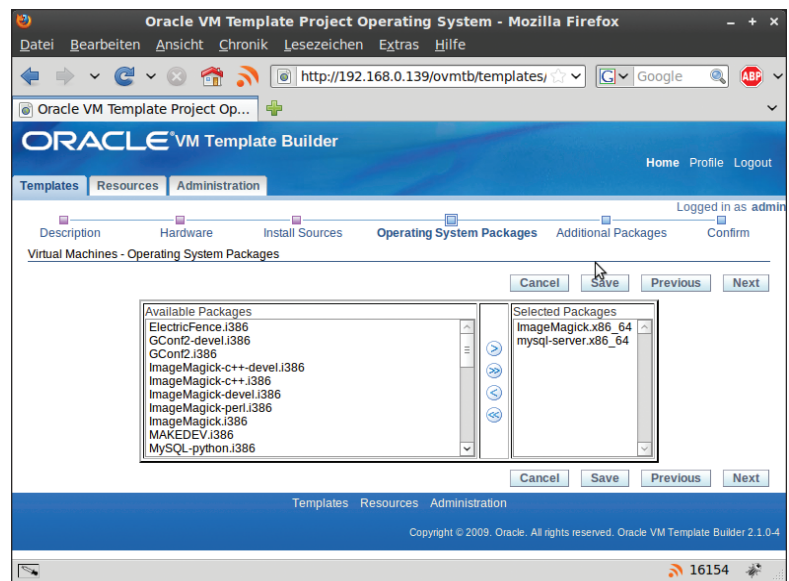


Abbildung 2: Die Auswahl zusätzlicher OEL-Pakete in Oracles Template Builder ist mit wenigen Klicks möglich.

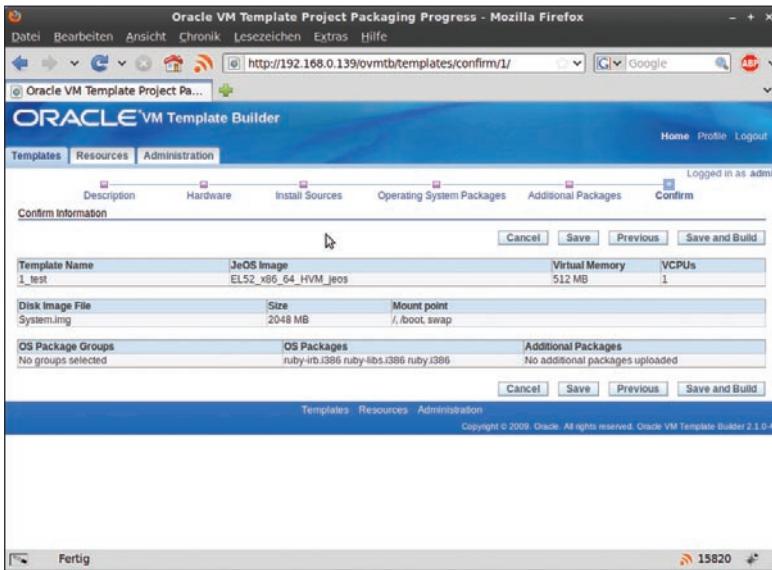


Abbildung 3: Der letzte Schritt vor dem Erstellen des Images: Zusammenfassung aller Parameter.

jeweils für 32-Bit oder 64 Bit, paravirtualisiert oder hardware-virtualisiert. Nach Eingabe des Template-Namens und der optionalen anderen Angaben legt der Anwender im nächsten Schritt die virtuelle Hardwareausstattung fest (Abbildung 1). Die könnte beispielsweise aus einer CPU bestehen, dazu 512 MByte RAM, 2 GByte Plattenplatz und ein Swap-Bereich von 512 MByte (Abbildung 1) Der RAM-Parameter stellt einen Maximalwert dar, im späteren Be-

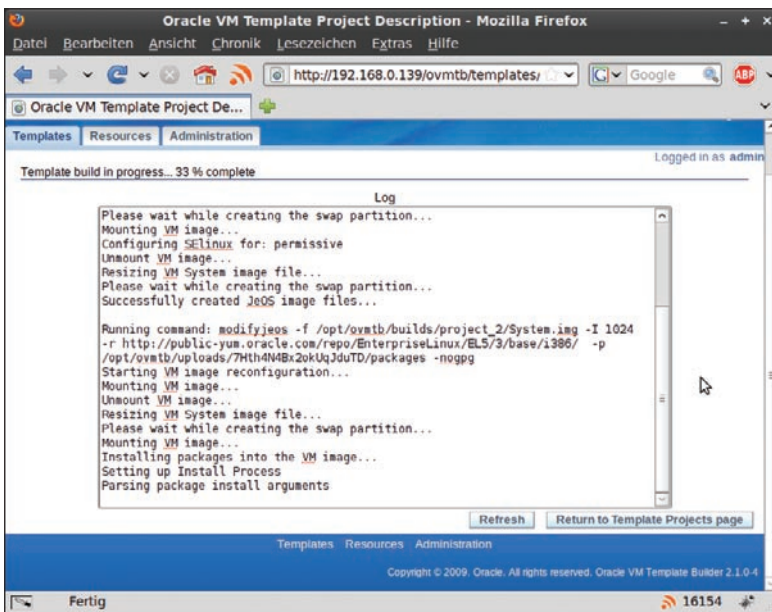


Abbildung 4: Während der Build-Prozess läuft, zeigt die Weboberfläche die Logmeldungen an.

trieb ist es ohne weiteres möglich, eine virtuelle Maschine auch mit weniger RAM zu starten.

Als Quellen für die Software kommen die Yum-Repositories der jeweiligen OEL-Version oder vorhandene ISO-Files in Frage. Hier ist allerdings darauf zu achten, das zum vorher ausgewählten JeOS-Image passende Repository zu wählen, da der Template Builder auch auch unpassende anbietet.

Im einfachsten Fall erzeugt man ein Template, das zusätzlich lediglich noch einige weitere Pakete aus dem OEL-Repository in ihrer Grundkonfiguration enthält (Abbildung 2). Paketabhängigkeiten werden automatisch aufgelöst. Alternativ kann auch ein vorhandenes ISO-Image als Quelle dienen.

Soll das System später beim Erstellen der virtuellen Maschine oder bei jedem Neustart beziehungsweise Herunterfahren konfiguriert werden, lassen sich im Template Builder die entsprechenden Shellskripte einbinden. Über das Erstellen dieser Skripte gibt der Template Builder Installation and User Guide im Abschnitt 7, „Creating Template Scripts“ ausführlich Auskunft. Anhang B des Guide zeigt einige Beispiele.

Konfigurationskripte für angepasste Systeme

Prinzipiell führt jedes Gast-System bei jedem Start das Skript »/etc/rc.d/init.d/oraclevm-template« aus. Beim ersten Start ruft dieses »init.d«-Skript einmalig »/usr/sbin/oraclevm-template --config« auf. Dieses Skript startet entweder eine interaktive Session zur Eingabe von Konfigurationsparametern oder – falls vorhanden – jenes Skript, das im Template Builder angegeben wurde. Als Default-Wert gibt der Template Builder »/u01/oraclevm-template.sh« vor, wobei »/u01« eine separate Festplatte ist. Existiert sie nicht, und ist auch kein anderer Pfad zu diesem Skript vorhanden, führt das System später keine zusätzlich definierten Konfigurationen aus.

Will man Anwendungen über separate Skripte konfigurieren, muss das Skript »oraclevm-template.sh« sie aufrufen. Der Template Builder erlaubt nur die Angabe von jeweils einem Skript für die Startup-Konfiguration und für das Cleanup beim Herunterfahren der virtuellen Maschine.

Sind alle Parameter für das zu erstellende Template vorhanden, listet der Wizard diese in einem abschließendem Screen zusammengefasst auf

(Abbildung 3) und übergibt die Konfiguration nach Bestätigung an eine Queue, die das Template im Hintergrund erstellt (Abbildung 4).

Das Deployment mit dem VM Manager

?

Noch Fragen?
 Unsere Autoren beantworten Ihre Fragen gern in unseren Foren.

Für das Deployment des fertigen Templates ist der Template Builder nicht mehr zuständig. Dies fällt stattdessen in den Aufgabenbereich des VM Managers, der in der Lage ist, vorhandene Templates auf allen unter seiner Verwaltung stehenden VM Servern zu erkennen.

Um ihm ein Template zur Verfügung zu stellen, muss der Anwender auf einem der VM Server im Pool ein Verzeichnis unterhalb von »/OVS/seed_pool« angelegen und die vom Template Builder erzeugten Image-Dateien (bei nur einer virtuellen Platte lediglich »System.img«) und die »vm.cfg« dorthin kopieren. Das kann er einfach mit »wget« erledigen, die URLs zeigt der Template Builder unter »URL for Manager« an. Der reale Pfad, etwa zum Kopieren mittels ssh oder rsync, lautet komplett »/opt/ovmtb/builds/project_name«

Sobald das Template auf einem VM Server vorhanden ist, kann man es im VM Manager registrieren (über »Resources|Import«, anschließend »Select from Server Pool« und über den Reiter »Approve« aktivieren).

Damit steht das Template zum Erzeugen von virtuellen Maschinen (im VM Manager über »Virtual Machines|Create Virtual Machines«, **Abbildung 5**) zur Verfügung. Schlägt der Start einer virtuellen Maschine fehl, zeigt der VM Manager ein Fehlersymbol an. Details sind in diesem Fall dann unter dem Reiter »Administration« in den Logs zu finden.

Fazit

Oracles Template Builder und VM Manager stellen jeweils einfach zu bedienende Oberflächen für das Anlegen und Verwalten von Server Templates zur Verfügung. Damit gelingt es, die Komplexität der dabei im Hintergrund tatsächlich ablaufenden Vorgänge wirkungsvoll zu verbergen. Auf der anderen Seite ermöglicht die vollständige Offenheit der Plattform nahezu

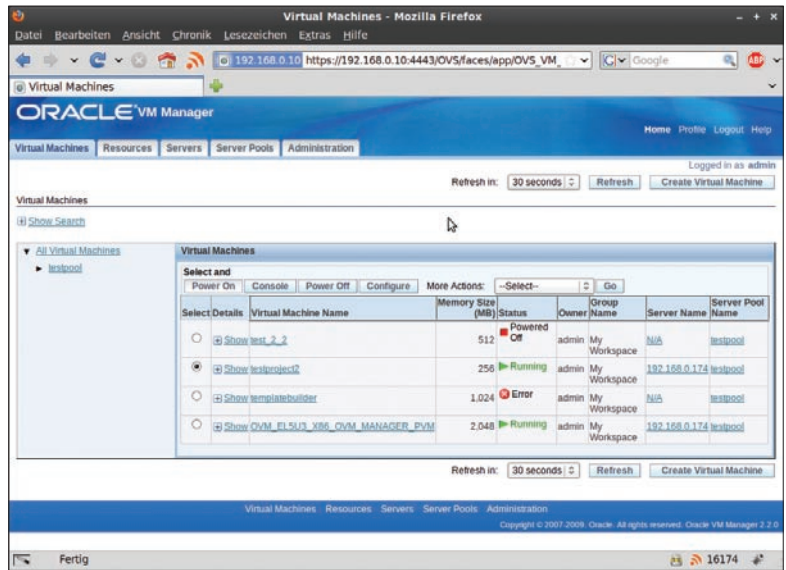


Abbildung 5: Neu erzeugte VMs im VM Manager. Dank der Templates sind sie mit wenigern Handgriffen fertiggestellt.

beliebige Anpassungen und Erweiterungen der VM Images.

Infos

- (1) Download der Oracle VM und deren Tools: (<http://edelivery.oracle.com/oraclevm>)
- (2) Dokumentation von VM Server, VM Manager und VM Template Builder: (http://download.oracle.com/docs/cd/E15458_01/index.htm)
- (3) Virtualisierungs-Whitepaper für ISVs, mit Schwerpunkt auf Server Templates: (<http://www.oracle.com/ocom/groups/public/@opnpublic/documents/webcontent/019929.pdf>)
- (4) Oracle JeOS: (http://www.oracle.com/technology/software/products/virtualization/vm_jeos.html)
- (5) VM Manager Command Line Interface; Beispiele für die Benutzung: (http://blogs.oracle.com/wim/2009/05/oracle_vm_manager_cli_and_web_1.html)
- (6) Oracle VM im Oracle Wiki mit ausführlicher Sammlung von technischen Dokumentationen: (<http://wiki.oracle.com/page/Oracle+VM>)

Der Autor

Ulrich Wolf ist einerseits freier Journalist, andererseits berät er Unternehmen beim Einsatz von Open-Source-Software.