

RED HAT

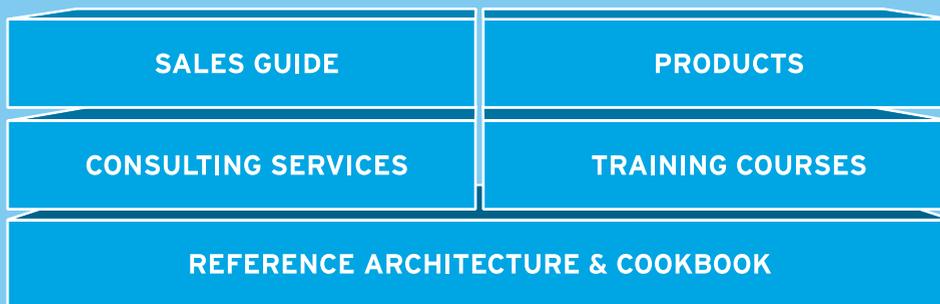
CLOUD FOUNDATIONS,

EDITION ONE: IMPLEMENTATION COOKBOOK



RED HAT CLOUD FOUNDATIONS, EDITION ONE

Real Clouds. Today.



This paper provides an overview of a reference architecture prepared by Red Hat that fully documents how to build your own private cloud. The publicly available, in-depth reference architecture captures the experience and knowledge of Red Hat engineers who went through the complete process of building a cloud. The reference architecture includes not only step-by-step instructions, but also scripts, configuration specifics, and port settings for cloud building. In short, it shows you how to implement a cloud starting from bare metal servers and ending with a complete, operating proof-of-concept. This document covers the highlights of Red Hat Cloud Foundations Reference Architecture, Edition One: Private IaaS (Infrastructure as a Service) Clouds. The full reference architecture document is at:

<https://inquiries.redhat.com/go/redhat/cloud-foundations>

Implementing the foundations for a private cloud are summarized in the following steps:

- **Step 1:** Install hardware and configure network.
- **Step 2:** Deploy highly available cloud management services.
- **Step 3:** Create cloud hosts and tenants.

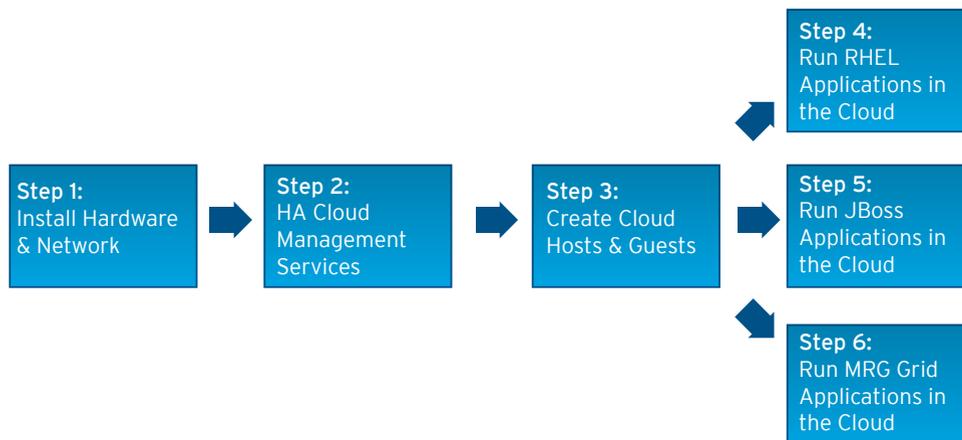
- **Step 4:** Run Red Hat Enterprise Linux applications in the cloud.
- **Step 5:** Run JBoss Enterprise Middleware applications in the cloud.
- **Step 6:** Run Red Hat Enterprise MRG Grid applications in the cloud.

In order to help you implement a private cloud, the following Red Hat courses and services are available to complement the Red Hat Cloud Reference Architecture:

- Red Hat Cloud Training Courses
- Red Hat Cloud Consulting Services

Future editions of the Red Hat Cloud Reference Architecture will develop the following concepts further:

- **Red Hat Cloud Foundations Reference Architecture:** Platform as a Service (PaaS)
- **Red Hat Cloud Foundations Reference Architecture:** Managing applications in the cloud
- **Red Hat Cloud Foundations Reference Architecture:** Managing mixed private clouds
- **Red Hat Cloud Foundations Reference Architecture:** Adding public clouds
- **Red Hat Cloud Reference Architecture:** Creating large-scale clouds



GLOSSARY

Cobbler: Installed with Red Hat Network (RHN) Satellite, Cobbler is an installation server that collects the various methods of performing unattended system installations, whether they are server or workstation systems.

Deltacloud: An application programming interface that abstracts the differences between various public and private clouds, thereby allowing workloads to be transportable.

KVM: Kernel-based Virtual Machine is a module of the Linux kernel that allows Linux to act as a Type 1 (native) hypervisor.

Red Hat Enterprise MRG: A next-generation IT infrastructure incorporating Messaging, Realtime, and Grid functionality – including support for a full spectrum of distributed tasks.

Red Hat Enterprise Linux (RHEL): Red Hat Enterprise Linux is the leading commercial Linux, which provides compatibility across private and public clouds.

Red Hat Enterprise Virtualization (RHEV): Refers collectively to the hypervisor and management components of Red Hat Enterprise Virtualization.

Red Hat Enterprise Virtualization-H (RHEV-H): Red Hat's "embedded hypervisor," which uses a tailored copy of Red Hat Enterprise Linux to provide a fully-compatible application environment in a small footprint.

Red Hat Enterprise Virtualization-M (RHEV-M): Red Hat Enterprise Virtualization-Management manages virtualized environments, including advanced features such as live migration.

Red Hat Network (RHN) Satellite: RHN Satellite provides management, provisioning, and monitoring that lets many servers be managed as one. Satellite serves as a software repository for an enterprise and delivers software updates and patches in a coordinated manner.

Tenant VM: A guest virtual machine used for a user workload.

OVERVIEW

The foundation for a private cloud described here illustrates a phased approach to technology adoption starting with server consolidation using virtualization, then adding services that help automate the deployment of additional software. By itself, virtualization offers many benefits to IT, including improved server utilization and the ability to move workloads around more easily. Expanding beyond virtualization helps an IT organization better manage the scale and complexity of their infrastructure.

This reference architecture also demonstrates how a cloud doesn't need to be built all at once but, rather, can build off a new or existing virtualization deployment. A proof-of-concept, such as this, is often applied first in a highly dynamic environment such as development and test, but it's a general purpose approach that can be applied to just about any use.

Red Hat's approach to cloud computing does not lock an enterprise into one vendor's cloud stack, but instead offers a rich set of products for building a cloud. These can be used alone or in conjunction with components from third-party vendors to create the cloud that best meets the customer's unique needs.

This reference architecture starts with the deployment of cloud infrastructure management services on a pair of servers configured as a cluster using Red Hat Cluster Suite (RHCS) for high availability. The infrastructure management services include:

- Red Hat Network (RHN) Satellite
- Red Hat Enterprise Virtualization Manager
- DNS and DHCP services
- RHCS management (using Conga)

Most of these services are installed in virtual machines (VMs).

The above cloud management services are used to configure a farm of Red Hat Enterprise Virtualization host systems and configure tenant VMs on the Red Hat Enterprise Virtualization hosts. Different use cases for deploying the following tenant VMs are described:

- Red Hat Enterprise Linux tenant VMs
- Windows tenant VMs

Cloud management services are used to deploy applications in the VMs. Specifically documented is the installation of the following types of application in tenant VMs:

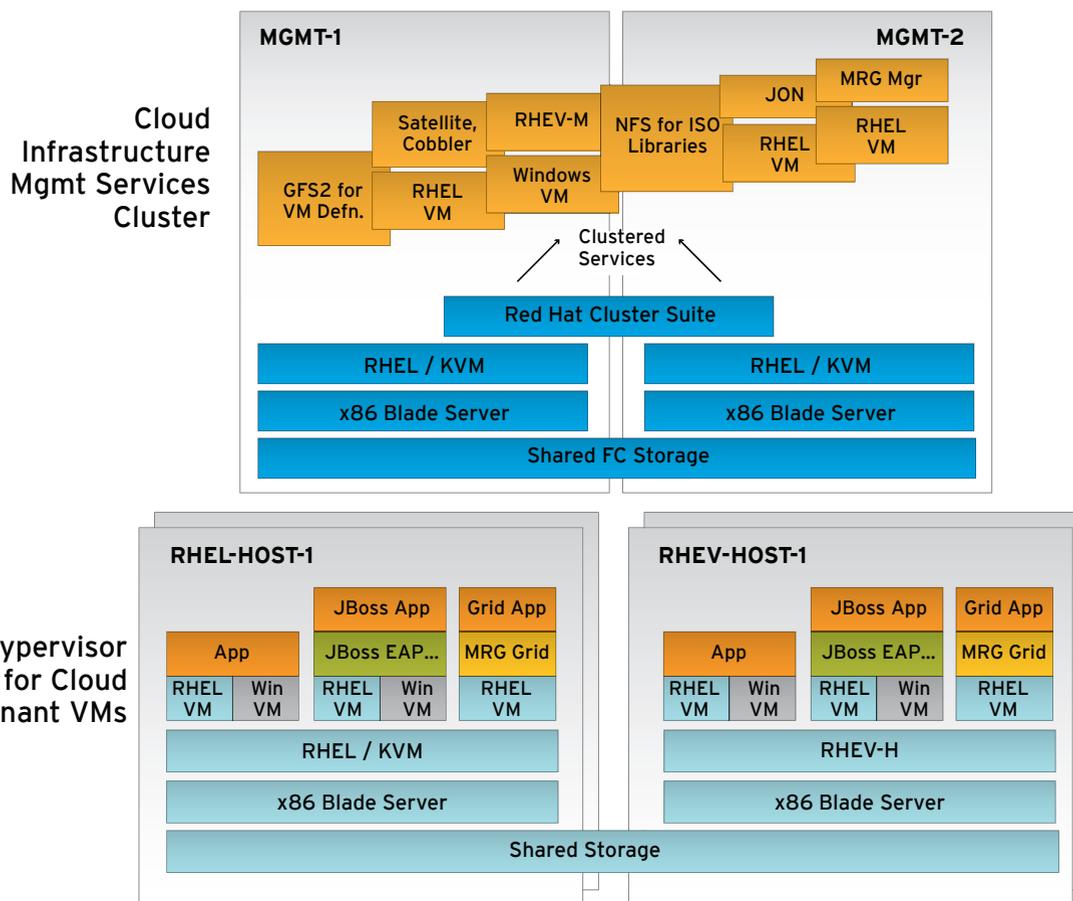
- Red Hat Enterprise Linux applications
- JBoss Enterprise Middleware applications
- Red Hat Enterprise MRG Grid applications.

This implementation cookbook summarizes key steps, identifies major components, and provides illustrative examples of deeper-level information, while the companion reference architecture document contains greater detail, including scripts and specific configuration settings.

STEP 1: INSTALL HARDWARE

The hardware foundation of this private cloud includes:

- A pair of server blades, connected to shared fibre channel disks and configured as a cluster for high availability, used to deploy cloud infrastructure management services. One of the servers is initially used to support the satellite server virtual machine (VM) that is in turn used to provision both management cluster members.
- A farm of server blades, connected to shared fibre channel storage, used to host the tenant VMs.



STEP 2: DEPLOY HIGHLY AVAILABLE CLOUD MANAGEMENT SERVICES

Cloud management services are used to set up the cloud hosts, the VMs within those hosts, and the applications in those VMs. To ensure the availability of these services, two Red Hat Enterprise Linux server blades are clustered (active/passive) using Red Hat Cluster Suite. Each of the cluster nodes is configured to run Red Hat Enterprise Linux with the bundled KVM hypervisor. For most management services, a VM is created using the KVM hypervisor and configured as an Red Hat Cluster Suite service. The management services, such as Red Hat Network (RHN) Satellite, are then installed in VMs. The Satellite VM provides the configuration management of the Red Hat Enterprise Linux system and is the network maintainer of services including Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), and Preboot eXecution Environment (PXE).

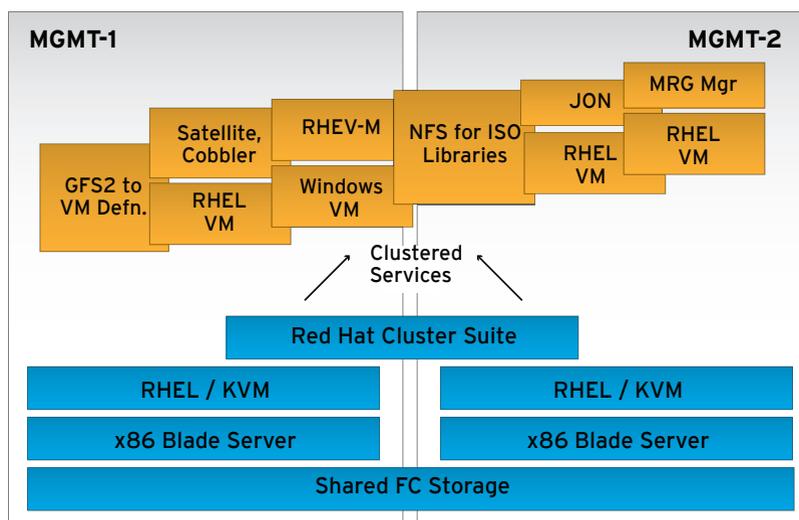
The first step is to provision one of the cluster member server blades that will create and temporarily host the satellite VM.

Install Python-based Expect scripts for communicating with vendor-specific components (Integrated Lights Out, On-board Administrator, Storage Array, and Virtual Connection Manager) on the server used to manage the installation.

Download the released Red Hat Enterprise Linux 5.5 DVD and duplicate the media. Modify the duplicate tree with a custom kickstart file configured to:

- install and configure the system
- copy the media to the installed system and configure its NFS export
- set the time and prepare the Network Time Protocol (NTP) daemon to auto start
- perform RHN registration
- configure preferred firewall settings
- convert the public network to a bridge
- configure resolv.conf
- disable the Large Receive Offload (LRO) feature

Cloud Infrastructure Management Services running in High Availability Cluster



The final step performed by the custom kickstart will prepare further actions for execution upon next boot of the satellite VM.

After the installation tree has been modified with the custom kickstart, create an ISO image from the modified directory, disconnect fibre channel storage, and install onto the server blade being temporarily used to create and configure the satellite VM.

Upon first boot, post-install scripts will:

- install the aforementioned Python-based Expect scripts for communicating with vendor-specific components
- configure the required volume groups for management services, cluster quorum, Global File System v2 (GFS2), and RHEV-M
- create the logical volume for the satellite VM.

Next the post-install scripts will:

- present the volumes to each host bus adapter (HBA) in the server
- deploy a device-mapper multipath configuration file preconfigured with recommended MSA array settings
- generate a secure shell configuration for distribution to other cluster members
- create the satellite VM
- remove the RHN registration of the temporary system to avoid conflict when this blade later registers as the second management cluster member

The creation of the satellite VM via custom kickstart not only installs the OS, it also:

- copies the resources directory to the system
- sets the time and prepares NTP to auto-start
- configures firewall rules
- registers the machine with RHN
- updates software
- configures DHCP and DNS
- performs Cobbler-related SELinux configurations
- deposits a preferred resolv.conf file
- prepares further actions to be performed upon next boot.

After the satellite software has completed installing, user interaction is required to add the first satellite user 'admin'.

Confirm that all of the required channels have been synchronized and execute a post-Satellite build script that will:

- create the Satellite infrastructure and tenant organizations and their default activation keys
- create activation keys for Red Hat Enterprise MRG systems
- provision the first management cluster member
- prepare satellite to install the JBoss Operations Network (JBoss ON) and Red Hat Enterprise MRG Manager VMs, as well as the Red Hat Enterprise Virtualization-Hypervisor and Red Hat Enterprise Linux/KVM hosts.

The provisioning of the first management cluster member (called from the post-Satellite build script) includes:

- setting member two as an NTP peer
- configuring *named* to act as a DNS slave
- configuring the first boot to host the satellite VM previously created on the temporary server
- deploying a preconfigured cluster configuration file
- creating a logical volume for use as Red Hat Enterprise Virtualization NFS storage for the ISO image library
- creating the other management VMs

Once the Satellite VM has started on the first management cluster member, its preconfigured rc.local commands will start the creation of the second management cluster member.

The creation of the second management cluster member includes post installation scripts that will:

- set member one as an NTP peer
- configure *named* to act as a DNS slave
- deploy the preconfigured cluster configuration file
- shut down the Satellite VM on member one
- label the quorum disk for cluster use

- add both interfaces of each member to the list of secure known hosts of the other member
- create and populate a GFS2 file system for VM configuration file storage
- configure and start cluster services on both members
- start the Satellite VM as a VM service

The cluster configuration file is preconfigured for cluster services including an NFS service providing storage for the Red Hat Enterprise Virtualization Management ISO image library, and VM services for Satellite, JBoss ON, Red Hat Enterprise MRG and Red Hat Enterprise Virtualization-Management.

Red Hat Enterprise Virtualization-Management provides enterprise virtualization management features including live migration, high availability, load balancing, power saving, and more. In this case, a Windows Server 2008 R2 VM is created that will require user interaction upon its first boot, and the installation of Red Hat

Enterprise Virtualization is achieved using the pre-downloaded installer.

The next step in setting up the management cluster is to use Red Hat Enterprise Virtualization-Management to create and configure a datacenter with a storage pool and a populated ISO image library for VM installations. This is done by creating a new datacenter, creating a new cluster within that datacenter, adding the pre-created Red Hat Enterprise Linux/KVM host, creating and attaching a storage pool, creating and attaching an ISO library, and finally populating the ISO library. The guest tools and Virt I/O driver images are already included in the predefined image used for installation.

Because the Red Hat Enterprise Virtualization-Hypervisor host cannot be installed until Red Hat Enterprise Virtualization-Management is operational, the final step is to install Red Hat Enterprise Virtualization-Hypervisor on the target server and reboot so that it can be discovered and approved in Red Hat Enterprise Virtualization-Management.

STEP 3: CREATE CLOUD HOSTS AND GUESTS

After creating the cloud infrastructure management services, the next steps involve creating virtualization hosts using either the Red Hat Enterprise Virtualization embedded hypervisor or Red Hat Enterprise Linux with KVM. This document uses Red Hat Enterprise Virtualization-H but the detailed reference architecture fully documents both cases.

PXE booting of the Red Hat Enterprise Virtualization-H live image is first enabled on the Satellite VM. This is done by downloading its RPM package from RHN, installing it, and using the included live CD tools to generate the files required for PXE. Then create profiles in Cobbler to automate the hypervisor configuration and installation process. After PXE booting the system, approve it from the Red Hat Enterprise Virtualization-M host tab. Tenant VMs can now be configured in a vari-

ety of different ways. Windows tenants can be deployed using either an ISO library or templates created from previously installed tenants. Red Hat Enterprise Linux tenants can use those two methods as well as PXE.

To install a Red Hat Enterprise Linux VM on a Red Hat Enterprise Virtualization-H host using PXE, an activation key is first created and configured as a user in the tenant organization of Satellite. This key is added to a new kickstart profile along with additional settings that help to automate the installation process. The new VM instance is started from Red Hat Enterprise Virtualization-M by specifying PXE in the boot sequence tab along with other details, such as the VM's memory size. After initial PXE booting, the cobbler PXE boot menu is displayed. Select the previously created kickstart.

Alternatively, if using the ISO library, first populate the ISO library with the Red Hat Enterprise Linux ISO image, which can be downloaded from the RHN website. This takes place through the ISO uploader utility in Red Hat Enterprise Virtualization-M. After mounting the ISO image as a DVD and setting it as the second device in the boot sequence, configure and boot the VM through Red Hat Enterprise Virtualization-M as before.

To use a template instead, prepare the template system to register with Satellite upon booting and clear hard-coded settings, such as hostnames; the `sys-unconfig` command is one way to make the system reconfigure network, authentication, and several other subsystems on the next boot. Then use the “Make Template” in

the Red Hat Enterprise Virtualization-M user interface. A new VM can now be created from this template by specifying the template’s name when creating a new VM from Red Hat Enterprise Virtualization-M. When the VM is ready to boot for the first time, register it with the Satellite VM because it will be known to Satellite with its parent ID until doing so.

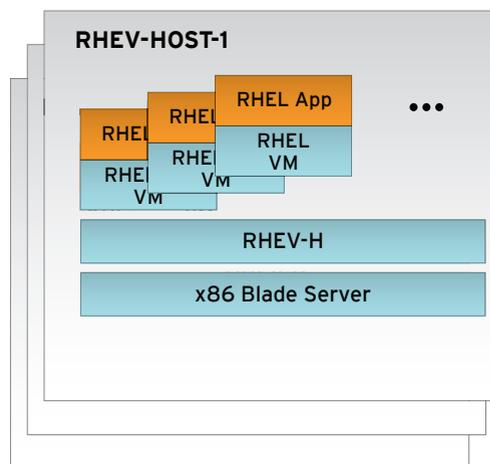
The main procedural difference for deploying Windows tenants using the ISO library or templates is that systems are prepared differently for templating with Windows. The details differ by Windows version, but the basic process involves using Microsoft’s System Preparation tool (`sysprep`) to strip the source VM of its individual computer identity.

STEP 4: RUN RED HAT ENTERPRISE LINUX APPLICATIONS IN THE CLOUD

The process with standard Red Hat Enterprise Linux applications is similar. They can be treated as what are sometimes called virtual appliances – effectively an application pre-combined with a tailored copy of the operating system. Alternatively, the operating system can be a standard or customized Red Hat Enterprise Linux image that is installed into a VM with the application layered on top in the usual manner.

Windows applications can also be installed into Windows VMs in this infrastructure.

Running RHEL Applications in Tenant VMs in the Cloud



STEP 5: RUN JBOSS ENTERPRISE APPLICATION PLATFORM APPLICATIONS IN THE CLOUD

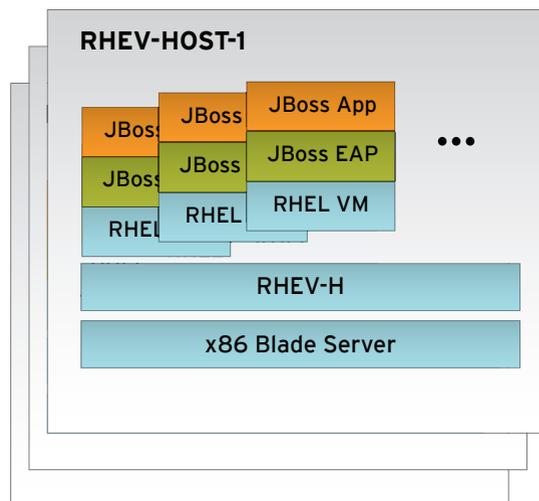
What has been covered so far takes the user through the process of creating a virtualized foundation for a cloud infrastructure. Building on top of that dynamic infrastructure can now begin. This section shows the process for JBoss Enterprise Application Platform (EAP) applications.

The first step is to install JBoss Operations Network, an integrated management platform that simplifies the development, testing, deployment, and monitoring of JBoss Enterprise Middleware; it gets installed in a Red Hat Enterprise Linux VM on the management services cluster.

Create a storage volume. Use Satellite to configure the activation key and create a new kickstart profile using the settings provided in the reference architecture. Next, run scripts available from Red Hat to configure firewall ports, update packages, and download JBoss ON. JBoss ON is then provisioned using the previously created kickstart profile. Now configure this VM as a cluster service in the same manner as the other VM services.

Applications and their associated middleware software stack can then be installed on the tenant VMs; the entire application stack can be PXE booted. After configuring kickstart, a post-installation script is used to open JBoss-specific firewall ports, ensure all installed software is up-to-date, install and configure the JBoss Enterprise Application Platform and JBoss ON agent, and deploy a JBoss application. The application can then be PXE booted in a VM using Red Hat Enterprise Virtualization-M. Once the installation is complete, the JBoss ON console dashboard will display the VM as an auto-discovered resource.

Running JBoss Applications in Tenant VMs in the Cloud



Because, in this example, each new JBoss VM automatically registers with JBoss ON, deployment via template will differ only as much as the VM created to act as a model for the template will not register itself with JBoss ON; this lets any VM created from this template register its own IP address. Using a template and a PowerShell script accessed through the Red Hat Enterprise Virtualization-M scripting library, multiple instances of a JBoss application VM can be rapidly provisioned on the tenant VM cluster. This allows for rapid scaling in response to changes in demand, a hallmark of a cloud architecture.

STEP 6: RUN RED HAT ENTERPRISE MRG GRID APPLICATIONS IN THE CLOUD

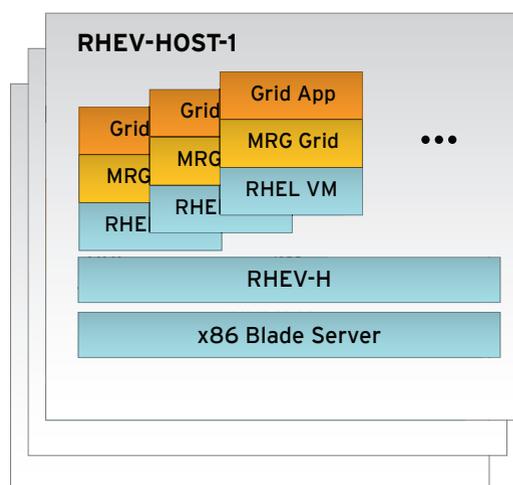
Red Hat Enterprise MRG Grid applications can also be installed on this platform. Based on the Condor project, MRG Grid provides the most advanced and scalable platform for high throughput and high performance computing with capabilities, such as:

- Advanced features for handling priorities, workflows, concurrency limits, utilization, low latency scheduling, and more.
- Support for a wide variety of tasks, ranging from sub-second calculations to long-running, highly parallel (MPI) jobs.
- The ability to schedule to all available computing resources, including local grids, remote grids, virtual machines, idle desktop workstations, and dynamically provisioned cloud infrastructure.

As with JBoss ON, the MRG Manager is installed as a cluster service on the management services cluster. It's installed via Satellite by preparing the MRG channels and configuring them with provided files that contain all non-default values. Create the storage area, configure the activation key, add required packages, and configure the kickstart. A supplied script performs the necessary configuration for the MRG Manage Console. After provisioning the MRG Manage VM, configure it as a cluster service.

As with the JBoss example, MRG Grid applications can be installed on the tenant system as part of the full software stack (Red Hat Enterprise Linux, MRG Grid middleware, plus the application itself).

Running MRG Grid Applications in Tenant VMs in the Cloud



TAKING IT FURTHER

Over time, Red Hat will add additional cloud services that build on and augment the components demonstrated in this proof-of-concept. However, as we have described, Red Hat today has the products necessary to configure a rich set of highly available management tools and services on top of a flexible, virtualized foundation. Applications installed into this environment can be rapidly resized, scaled, updated, or retired as requirements change.

In short, this is an on-ramp for customers who want to either dip their toes into cloud computing or want to begin building a more ambitious implementation.

Red Hat provides not only the products to build a cloud implementation, but also the documentation, scripts, detailed step-by-step instructions, and services to get started today.

Future versions of the reference architecture will describe expanded details of Red Hat's cloud infrastructure, including the following:

- Managing applications in the cloud
- Managing mixed private clouds
- Adding public clouds
- Creating large-scale clouds

RED HAT CLOUD TRAINING COURSE

Red Hat Training provides a robust set of technologies and partnerships for customers to learn to build cloud solutions. Among the key technologies are Red Hat Enterprise Virtualization, Red Hat Network Satellite, Red Hat Cluster Suite, JBoss Enterprise Application Platform, and JBoss Operations Network. Red Hat Training courses are available for each of these technologies today. The cloud training course will leverage Red Hat's Cloud Reference Architecture and bring these technologies together into an integrated training course on how to build an enterprise cloud solution with Red Hat.

The course (available Fall 2010) is targeted at:

- Senior system architects, administrators, consultants, and other IT professionals responsible for the planning and implementation of cloud solutions
- IT professionals at cloud providers wanting to understand how to use Red Hat technologies to build enterprise-class cloud solutions

RED HAT CLOUD CONSULTING SERVICES

Red Hat Consulting delivers services to help you navigate and execute your cloud adoption strategy:

I. Red Hat Cloud Quickstart

For customers ready to explore the viability of cloud in their infrastructure and develop proof points of cloud viability.

II. Red Hat Services Pathway to Cloud Computing

For customers ready to embark on a path to strategic planning and deployment of cloud technologies.

I. RED HAT CLOUD QUICKSTART

The Red Hat Cloud Quickstart consulting engagement leverages Red Hat's unique business and technology cloud model to help customers get started with a prototype cloud infrastructure.

The engagement provides on-site setup and configuration of cloud infrastructure technologies (up to ten servers) and knowledge transfer. For customers ready to develop proof points of cloud viability, the quickstart can be coupled with a migration of one to two actual

workloads to the established test cloud to deliver a proof-of-concept.

The engagement helps organizations accelerate time-to-deployment, with on-site product experts delivering configuration, support, and knowledge transfer. At the end of the engagement, customers can assess the viability of cloud for their enterprise and, if chosen, can demonstrate a tangible implementation of a private cloud for testing purposes. This is a valuable step in cloud adoption, providing confidence in the organization's ability to support larger cloud strategies.

Engagement duration:

Quickstart: Two weeks

Quickstart plus proof-of-concept: Four weeks

Deliverables:

Sets up a Red Hat virtualization environment along with JBoss Enterprise SOA Platform.

- Install and configure Red Hat Enterprise Virtualization Manager for Servers and two Red Hat Enterprise Virtualization Hypervisor hosts.
- Install two Linux virtual machines into the Red Hat Enterprise Virtualization environment.

- Prepare IT staff with the knowledge to manage virtualization tasks beyond engagement. Includes a one-day Red Hat Enterprise Virtualization Manager Workshop.
- Install JBoss Enterprise SOA Platform onto each virtual machine, along with a demonstration service.
- Document the setup and create an OVF image of the virtualized environment.
- Provide SOA Workshop on creating services on JBoss Enterprise SOA Platform.

II. RED HAT SERVICES PATHWAY TO CLOUD COMPUTING

The Red Hat Services Pathway to Cloud Computing is a series of strategic consulting engagements that develop and implement a tailored roadmap to help customers move into a future of cloud computing and business agility.

With Red Hat's Cloud Computing Pathway, organizations can build a private or public cloud infrastructure now using open standards as well as proven designs

and configurations. The Pathway to Cloud Computing leverages open source technologies, tools and standards, industry insight, and proven IT methodologies to provide you with the best approach to cloud computing.

Each Pathway to Cloud Computing consists of four core strategic engagements:

- **Cloud Readiness Assessment:** Analyzes the existing environment and provides a high-level roadmap to cloud adoption.
- **Cloud Quickstart:** Leverages Red Hat's unique business and technology cloud model to help customers get started with a prototype cloud infrastructure.
- **Cloud Architecture:** Provides recommendations on which workloads are best suited for cloud.
- **Cloud Implementation:** Builds a test bed cloud infrastructure.

Engagement duration

Red Hat Services Pathways are tailored to each customer's needs. As a result, full pathway durations will vary. The first step along the Pathway to Cloud Computing, the Cloud Readiness Assessment is typically a four-week engagement.



Copyright © 2010 Red Hat, Inc. Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, and RHCE are trademarks of Red Hat, Inc., registered in the U.S. and other countries. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.



For more information about Red Hat and cloud computing,
visit www.redhat.com/solutions/cloud.