

# Construction and Management Implementation of Virtual Machine System on Containerized Platform

Wenke Xu

Yianxin Information Technology R&D (Shanghai) Co., Ltd., Shanghai, 200000, China

## Abstract

This paper aims to provide a detailed introduction to a virtual machine operation and maintenance management and implementation plan based on a containerized platform. The system aims to provide an efficient and convenient container based virtual machine operation and maintenance solution to meet the needs of enterprise business transformation. By analyzing the feasibility and functional requirements of the system, we have delved into the construction, management, and operation of virtual machines on containerized platforms, the implementation details of migrating virtual machines to containerized platforms, the implementation of dependency management for external consumption services of virtual machines, and the network configuration for providing external services through clusters between virtual machines. This paper summarizes the design and implementation process of the system, emphasizing the practicality of providing containerized platform virtual machine construction management in daily enterprise operations. Through the application of this system, enterprises can easily migrate virtual machines to containerized platforms and coexist with other container based services, thereby improving the overall operation and maintenance experience, improving operation and maintenance efficiency, and reducing costs.

## Keywords

virtual machine; virtualization; migration; containerized platform

## 容器化平台虚拟机系统的构建管理实施

徐文可

易安信信息技术研发（上海）有限公司，中国·上海 200000

## 摘要

论文旨在详细介绍一个基于容器化平台上的虚拟机运维管理与实施方案。该系统旨在提供一种高效、便捷的容器化平台的虚拟机运维方案，以满足企业业务转型需求。通过分析系统的可行性和功能性需求，我们深入探讨了在容器化平台进行虚拟机的构建管理运维，迁移虚拟机到容器化平台的实现细节，虚拟机对外消费服务依赖管理实现，以及虚拟机之间透过集群提供对外服务的网络配置。论文总结了系统的设计和实施过程，强调了提供容器化平台虚拟机构建管理在企业日常运营中的可实践性。通过该系统的应用，企业能够可以方便地把虚拟机迁移到容器化平台，并与其他基于容器的服务共存，从而提升整体运维体验，提高运维效率，降低成本。

## 关键词

虚拟机；虚拟化；迁移；容器化平台

## 1 引言

虚拟机是公司日常运维的主要载体之一，随着越来越多云业务的转型需要，客户在 vsphere 平台使用 VM 的成本变化，大多数企业容器化转型的需求加大。为了验证虚拟机转向容器化平台的可能性，我们寻找一种便捷、高效的解决方案，可以帮助实施从 vsphere 平台传统虚拟机轻松迁移到值得信赖、一致且全面的混合云应用平台上，并对其管理。为此，我们采用基于红帽® OpenShift® 的虚拟化集群的实现，它为企业组织运行和部署新的和现有的虚拟机（VM）工作

负载提供了一个现代化平台。在论文中，我们将深入研究这些模块的实现细节，介绍系统架构、技术手段和关键功能。

## 2 项目概述

红帽® OpenShift® 虚拟化是利用云原生应用平台的简易性和速度优势，保留现有的虚拟化投资，同时采用现代管理原则，为企业组织运行和部署新的和现有的虚拟机（VM）工作负载提供了一个现代化平台。借助该解决方案，用户可将传统虚拟机轻松迁移到值得信赖、一致全面的混合云应用平台上。本项目旨在为运行传统 vm 的客户提供一套基于容器化平台的解决方案，高效地管理虚拟机。本项目的主要目标是实现虚拟机的正常构建，实施虚拟机迁移，日常管理等功能需求。具体包括以下方面：容器化平台上 VM 的构建，实时

【作者简介】徐文可（1991-），女，中国江苏泰兴人，本科，工程师，从事软件测试研究。

迁移,日常运维管理访问监控。虚拟机从传统平台迁移到容器化平台。迁移在容器化平台的正常消费,以及虚拟机在平台上网络配置访问。

### 3 系统需求分析

#### 3.1 系统的可行性分析

第一,技术可行性。一方面,确定能满足用户商业需求的计算,存储,网络功能的容器化平台。另一方面,评估所需的服务器、网络设备和存储设备等硬件资源,以支持系统的正常运行。确保硬件资源的可获得性和性能能够满足预期的负载。第二,经济可行性。一方面,估算项目的总成本,包括软件开发、硬件采购、培训和维护等各个方面的费用。同时也需要估算系统的运行成本,包括服务器维护、人员培训和数据存储成本。另一方面,预测项目的预期回报,包括收入增加、运营效率提高和客户满意度的提升等。这些预测需要基于市场调研和竞争分析来进行。

#### 3.2 系统功能性需求分析

系统功能性需求分析是容器化平台虚拟机管理运维的关键步骤之一。在这一阶段,将明确系统应该具备哪些功能,以满足容器化平台上虚拟机的构建和管理的各种需求。第一,要确定支持虚拟机运行的容器化平台。平台需要能够提供虚拟化的支持。例如,在容器化平台上虚拟机进行快照,关机,开机,克隆。不同节点之间实时热迁移。第二,平台支持虚拟机迁移。不同操作系统的虚拟机迁移,都能够在成功迁移后继续正常运行。迁移必须能支持冷迁移,即关闭虚拟机之后迁移到容器化平台。第三,用户的外接服务在迁移之后可以继续。比如模拟用户在消费外部的数据库系统比如mysql,在迁移虚拟机或者节点间实时迁移起来之后能正常消费。第四,网络管理模块。基于平台可以实现能够提供虚拟机之间的网络服务,虚拟机能够通过互相通信,并通过集群IP对外提供服务。

### 4 系统功能实现

#### 4.1 虚机构建实现

平台能够支持虚拟化的实现是关键,OpenShift Virtualization是OpenShift Container Platform的一个附加组件,它可以用于运行和管理虚拟机工作负载以及容器工作负载。OpenShift Virtualization源于CNCF的社区项目KubeVirt,由红帽主要贡献,并进行针对性优化。KubeVirt是Redhat开源的以容器方式运行虚拟机的项目,以k8s add-on方式,利用k8s CRD为增加资源类VirtualMachineInstance(VMI),使用容器的image registry去创建虚拟机并提供VM生命周期管理。

OpenShift虚拟化,在KubeVirt的基础上,为企业客户提供生产就绪环境所需要的功能和性能加固和增强,并配合OpenShift良好生态,包括裸机管理、GPU直通、软件定义网络、OpenShift容器原生存储。它提供了在常规容器化工作负载旁边运行虚拟机的能力。OpenShift虚拟化安装还

可以采用OpenShift OperatorHub软件商店的方式,进行一键式安装和部署。

#### 4.2 VM的创建与管理

为了实现对VM的构建与管理,在容器化平台准备完毕后,可以从operator hub中选择在线安装OpenShift Virtualization Operator,然后配置创建hyperconverged对象。

具体安装方式是多样化的。可以用线上提供的模板,或通过上传ISO文件到OCP平台上的PVC来启动安装过程,也可以使用命令行的方式来实现VM的构建。举例如:在OCP平台的Web UI的中选择对应操作系统的模板来快速创建如fedora。在使用指定模板的同时,支持对模板的部分参数修改来自定义用户名,密码,磁盘容量,网卡类型等。特别地,针对Windows server这一类型的vm,可以通过创建PVC的方式,将ISO上传至新建好的PVC,boot到OS安装界面启动。在虚拟机安装完毕之后,使用web UI console或者通过virtctl命令的方式来远程访问vm,并且可用web UI提供的clone功能来完成快速批量克隆。

对已经构建的VM,选择在不同的多个时间节点进行快照,回滚,VM的运行始终保持正常。连接外部的mysql server并且正常消费mysql服务之后,在OCP平台执行虚拟机的live migration。虚拟机live migrate到OCP集群其他物理节点后,能正常启动并且持续消费mysql server。通过ping命令观察迁移中的实时迁移的连接中断时间小于1s,连接中断时间很短。

#### 4.3 VM迁移的实现

Migration Toolkit for Virtualization(MTV) operator的实现是能进行VM迁移实现的关键。Migration Toolkit for Virtualization是RedHat提供的一个开源项目,使用它可以VMware和KVM的虚拟机迁移到OpenShiftVirtualization环境中运行。MigrationToolkitforVirtualization支持Cold和Warm两种迁移,本次着重介绍的是冷迁移过程。

首先配置以下环境,准备源主机,比如一个部署VM的ESXi host,VM操作系统可以是windows server 2019,ubuntu20.24等。目标主机是:OCP集群平台。借助在OCP集群提供的operator hub,在线安装Migration Toolkit for Virtualization operator(MTV)。通过页面来实现用于执行VM的并行迁移。其主要实现过程包含以下:①从vsphere下载并build一个VDDK镜像推送到OCP的registry server中,此镜像在此场景中主要用于加速VM的迁移,缩短单个VM在转移disk这一过程所消耗的迁移时间。②VM provider配置。该配置主要用于连接访问目标源主机上所需要的vsphere登录信息。③Migration plan配置,用于创建具体执行vm的迁移计划,主要包括选择指定迁移的VM,读取并显示迁移前的网络和存储配置信息,指定迁移之后在容器化平台可以使用的网络模式,可以使用的存储类型。目前OCP平台使用对象存储或者块存储都可以满足VM对

存储的消费要求，VM 都可以基于这样的存储正常运行。④ 执行迁移。在 migrate plan 创建完成后，即可 web UI 上选择点击开始执行迁移计划，首先 validate 输入，并确认 VDDK image 是否可用，之后开始实时，页面展示具体的迁移进度细节。当配置需要更新，可以 cancel migrate plan，重复上述三步创建新的 migrate plan。

## 4.4 虚拟机外接服务访问实现

### 4.4.1 服务搭建实现

数据库是在 VM 的日常运行中经常用到的对外访问的服务，因此，需要考虑迁移前后，VM 对外消费服务是否可以继续正常工作。本项目中采用关系型数据库来存储信息，比如准备一个 VM，安装使用 mysql server 并生成 DB 和表等信息，对外提供数据库的访问。

### 4.4.2 消费 mysql 服务

举例以下场景：准备一个 python 脚本，实现如下功能：连接外接的数据库服务器，读写数据记录，持续进行增删改查的操作，模拟实际用户场景下对数据库的消费。通过添加超时连接重试机制来模拟网络中断等可能造成失败重连。同时，引用日志模块并打印生成实时日志来追踪每次读写记录操作的执行时间。

在 vsphere 平台上部署 VM 使用该脚本，实现数据库的连接，并将此服务设置成开机自启动。在执行对外部数据库的正常访问后，执行对 VM 的迁移完成后，查看服务状态并查阅日志的访问时间，在迁移过程之后查看虚拟机仍然能够正常去读取数据库记录。迁移过程中 session 会发生中断，可被超时连接重试机制捕获，保证外接消费服务的不间断。

## 5 VM 的网络配置管理实现

缺省情况下运行在平台中的 VM 使用的是 Pod 网络，同时 VM 迁移时默认也是 Pod Networking 配置。这样只能通过 Pod 网络实现从 OpenShift 集群外部访问 VM 或从 VM 访问外部资源。由于 VM 重启对应的 Pod 都会变更 IP，如果只需要从集群内部访问 VM，使用 service 对应 pod，但如果需要从 Openshift 集群外部访问 VM，需要为 service 绑定 Node Port。

为了让 VM 能够直接访问外部集群或使用集群 IP 对外提供服务，可以采用 Kubernetes NMState Operator。Kubernetes NMState Operator 提供了一个 Kubernetes API，用于使用 NMState 在 OpenShift Container Platform 集群的节点上执行状态驱动的网络配置。Kubernetes NMState Operator 为用户提供了在集群节点上配置各种网络接口类型、DNS 和路由的功能。另外，集群节点中的守护进程会定期向 API 服

务器报告每个节点的网络接口状态。

OCP 集群平台的 OperatorHub 中提供 NMstateOperator 的安装。在线安装完成执行以下步骤。首先，创建一个 NNCP 对象。可以选择指定节点上 up 的一张未被使用的指定网卡，基于这个网卡名称创建 Node Network Configuration Policy 对象，用来描述节点上请求的网络配置，它通过将 Node Network Configuration Policy 清单应用到集群来更新节点网络配置，包括添加和删除网络接口。

其次，创建 Network Attachment Definition (NAD)，这是用于定义如何将网络接口附加到容器运行时的方法。在 Create Network Attachment Definition 页面下创建 Network Attachment Definition 时可选择使用 LinuxBridge/VLAN/Overlay 模式。

最后，在 VM 中修改默认的网卡或者新增上面创建的网络配置。对 ubuntu 等 linux vms 添加指定创建网卡。在上述使用默认的 DHCP 模式配置下 VM 可以自动获取 OCP 集群分配的 IP。由此实现构建的不同操作系统 VM 互相通信，并可以提供对外暴露虚拟机内自带的服务实现。

## 6 结语

基于容器化平台的 VM 的迁移与管理在企业运维运营中具有重要意义。通过对 VM 的迁移与管理，成功地兼容了传统 VM 在 K8S 平台上的运营与运维。管理员可以更轻松地管理构建、管理虚拟机，一键部署迁移，基于容器化平台对外通过日常基本服务。然而，我们也要认识到系统运维中可能出现的挑战。目前支持的 VM 并不包含足够多的版本的支持，以及有些源 VM 处在复杂网络下，VM 能否显示在迁移列表中。在迁移成功后，基于源主机的 IP 地址等信息在容器化平台上是否可以全部保存。另外，考虑用户对平台能提供的容灾备份的需求，针对复杂网络的具体场景实现，以及后续的安全性、性能优化和持续维护仍然需要进一步细化。在系统的进一步演化中，要不断优化，以适应业务的不断变化。总之，能在容器化平台上进行 VM 的构建，迁移，对外服务，有望改善企业运维效率，降低运维成本开销。

### 参考文献

- [1] <https://kubevirt.io/user-guide/architecture>.
- [2] [https://access.redhat.com/documentation/en-us/openshift\\_container\\_platform/4.14/html-single/virtualization](https://access.redhat.com/documentation/en-us/openshift_container_platform/4.14/html-single/virtualization).
- [3] [https://access.redhat.com/documentation/zh-cn/migration\\_toolkit\\_for\\_virtualization/2.6/html/installing\\_and\\_using\\_the\\_migration\\_toolkit\\_for\\_virtualization/index](https://access.redhat.com/documentation/zh-cn/migration_toolkit_for_virtualization/2.6/html/installing_and_using_the_migration_toolkit_for_virtualization/index).
- [4] <https://github.com/nmstate/kubernetes-nmstate>.