

云计算与虚拟化技术



第10讲：Templates、vApps

阮晓龙

13938213680 / rxl@hactcm.edu.cn

<http://cloud.xg.hactcm.edu.cn>
<http://www.51xueweb.cn>

河南中医药大学信息管理与信息系统教研室
信息技术学院网络与信息系统科研工作室

2019.5

讨论提纲

- Cloning VMs
 - Creating a Customization Specification
 - Cloning a Virtual Machine
 - Introducing vSphere Instant Cloning
 - Instant Cloning Running VMs
 - Instant Cloning Frozen VMs
- Creating Templates and Deploying Virtual Machines
- Using OVF Templates
- Using Content Libraries
- Working with vApps



1. Cloning VMs

1.1 Creating a Customization Specification

- ❑ vCenter Server's customization specification works hand in hand with the tools for customizing VM clones (**Sysprep** for VMs with a Windows-based guest OS; **open source tools** for a VM with a Linux-based guest OS).
- ❑ you have to provide vCenter Server with the information necessary to give the cloned VM its own unique identity. This includes the IP address, passwords, computer name, and licensing information. With customization specification, you provide all the information only once and then apply it as needed when cloning a VM.



1. Cloning VMs

1.1 Creating a Customization Specification

- Create a customization specification in the following two ways:
 - By creating it during the process of cloning a VM
 - By using the Customization Specification Manager in vCenter Server



1. Cloning VMs

1.1 Creating a Customization Specification



- Demo on vCenter Server
 - vSphere Web Client
 - Create a new customization specification for Windows



1. Cloning VMs

1.2 Cloning a Virtual Machine



- Demo with vSphere Web Client
 - step by step to clone a VM
 - Cloning Running VMs
 - It's possible to clone even powered-on VMs.



1. Cloning VMs

1.3 Introducing vSphere Instant Cloning

- Creating a clone of a VM in a matter of minutes can be very helpful, but if you want to create a large number of identical VMs, the cloning process can take hours.
 - Let's say your job involves administering a pool of virtual desktops in a virtual desktop infrastructure (VDI) farm, or maybe you are spinning up a large number of VMs for developers to work on an exact replica of the main production environment.
 - VMware has enhanced the traditional cloning capabilities to increase the speed and reduce the resource overhead when there's a need to create multiple versions of a source VM at the same time.



1. Cloning VMs

1.3 Introducing vSphere Instant Cloning

- It's easy to derive from its name that instant cloning is fast, but just how fast is it?
- That depends on a few things, such as:
 - the ESXi host speed, utilization, and the size of the source VM, but usually it is somewhere between one and a few seconds.
 - Not only can it create VMs quickly, but it can do it in parallel too—up to eight VMs can be cloned simultaneously!
 - In a best-case scenario, you could generate 100 brand new identical VMs in a 10- to 20-second time frame.



1. Cloning VMs

1.3 Introducing vSphere Instant Cloning

- The Instant Clone feature will create VMs that run in the exact state their source VM does by leveraging a few technologies: vMotion, Transparent Page Sharing, and delta disks.
 - With traditional VM cloning, a complete set of the source VM's virtual hardware is created.
 - You get a copy of the all virtual disk (.VMDK) files, a copy of the VMs configuration (.VMX) files, and even a copy of the in-guest memory (if the cloned VM is currently running).



1. Cloning VMs

1.3 Introducing vSphere Instant Cloning

- The Instant Clone feature will create VMs that run in the exact state their source VM does by leveraging a few technologies: vMotion, Transparent Page Sharing, and delta disks.
 - The Instant Clone works is completely different.
 - The result is more or less the same, in that you still get a new VM, but you don't get brand new copies of the virtual disks or in-guest memory for running VMs.
 - Instant Clone uses a function known as copy-on-write to share both memory and disk data with the source VM, and only making a copy of the changes when the cloned VM needs to modify something.
 - That way, it can keep a single copy of all data without needing to duplicate anything, and it can create new VMs based on the same original VM very quickly.



1. Cloning VMs

1.3 Introducing vSphere Instant Cloning

- The Instant Clone feature will create VMs that run in the exact state their source VM does by leveraging a few technologies: vMotion, Transparent Page Sharing, and delta disks.
 - There are two different ways to instantly clone a virtual machine.
 - The source VM can be in a powered-on and running state.
 - The source VM can be set to a special powered-on and “frozen” state.

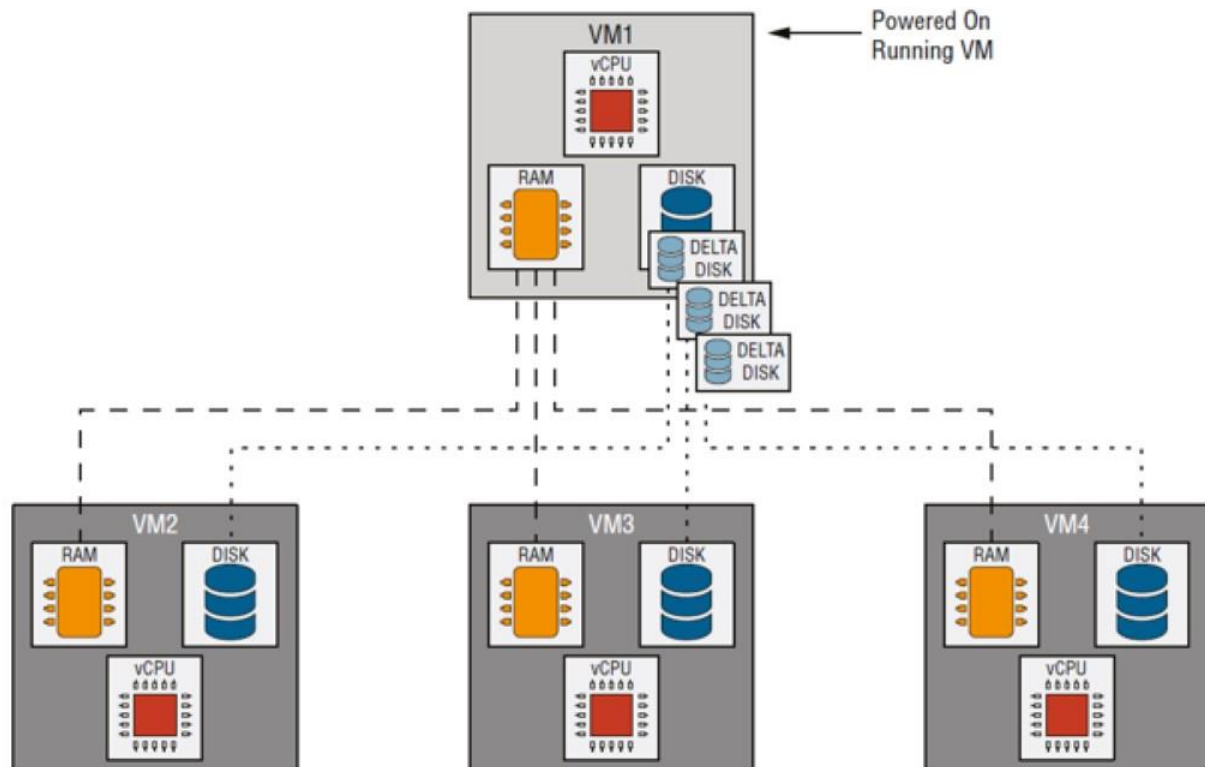


1. Cloning VMs

1.4 Instant Cloning Running VMs

FIGURE 10.8

The instant-clone process for powered-on VMs creates a new delta disk on the source VM each time a clone is created.

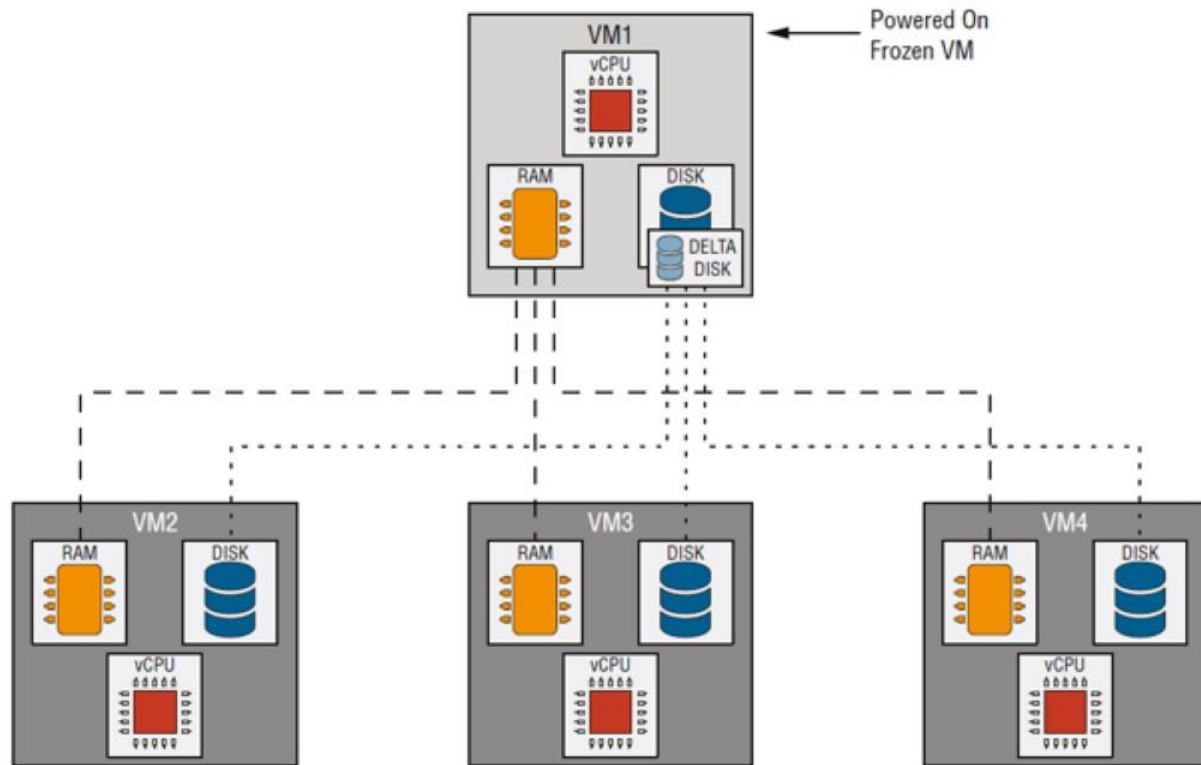


1. Cloning VMs

1.5 Instant Cloning Frozen VMs

FIGURE 10.9

Frozen VMs will never have more than a single delta disk and can be cloned as many times as needed.





- Demo with vSphere Web Client
 - Instant Cloning Running VMs
 - Instant Cloning Frozen VMs

2.Creating Templates and Deploying Virtual Machines

2.1Cloning a Virtual Machine to a Template

- vCenter Server offers four options for creating templates:
 - Clone To Template
 - Clone To Template In Library
 - Convert To Template
 - Export OVF Template
- In all cases, you'll start with a VM that already has an instance of a guest OS installed.



2.Creating Templates and Deploying Virtual Machines

2.1Cloning a Virtual Machine to a Template

- The Clone To features copy this initial VM to a template format or another VM, leaving the original VM intact.
- The Convert To Template feature takes the initial VM and changes it to template format, thereby removing the ability to perform power operations on the VM without converting back to VM format.
 - Using either approach, once the VM is in template format, that template cannot be powered on or have its settings edited.
 - It's now in a protected format that prevents administrators from inadvertently or unintentionally modifying the "gold image" from which other VMs are deployed.



2. Creating Templates and Deploying Virtual Machines

2.1 Cloning a Virtual Machine to a Template

- When considering which VMs you should convert to templates, remember that the idea behind a template is to have a pristine system configuration that can be customized as needed for deployment to the target environment.
 - Any information stored on a VM that becomes a template will become part of the new system deployed from that template.
 - If you have VMs that are critical servers for production environments with applications installed, those are not good candidates for templates.
 - The best VMs to use for templates are VMs with a new, clean installation of the guest OS and any other base components.
 - At a minimum, you should always install VMware Tools.



2.Creating Templates and Deploying Virtual Machines

2.1Cloning a Virtual Machine to a Template

- In fact, we recommend creating a new VM specifically for use as a template or creating the template from a VM as soon after creation as possible.
 - This ensures that the template is as pristine as possible and that all VMs cloned from that template will start out the same way.
 - You can convert a VM to a template using the context menu of the VM or the Convert To Template link in the Commands list.
 - Because templates cannot be modified, to make changes or perform updates to a template you must first convert the template back to a VM, then update it, and finally convert it back to a template.
 - Note that the Convert To Template command is grayed out if the VM is currently powered on.
 - To use the Convert To Template command, the VM must be powered off.



2. Creating Templates and Deploying Virtual Machines

2.1 Cloning a Virtual Machine to a Template



- Demo with vSphere Web Client
 - Cloning a Virtual Machine to a Template



2. Creating Templates and Deploying Virtual Machines

2.2 Deploying a Virtual Machine from a Template



- Demo with vSphere Web Client
 - Deploying a Virtual Machine from a Template



3.Using OVF Templates

- ❑ Open Virtualization Format (formerly called Open Virtual Machine Format) is a Distributed Management Task Force (DMTF) standard format for describing the configuration of a VM.
- ❑ Although it was originally pioneered by VMware and other industry contributors, most virtualization vendors now support OVF as well.
- ❑ VMware vSphere 6.7 provides OVF support in three ways:
 - Deploying new VMs from an OVF template (essentially, importing a VM in OVF format)
 - Exporting a VM as an OVF template
 - Storing OVF templates within a Content Library



3.Using OVF Templates

3.1Deploying a VM from an OVF Template



- Demo with vSphere Web Client
 - Deploying a VM from an OVF Template



3.Using OVF Templates

3.2Exporting a VM as an OVF Template

- vCenter Server lets you export an existing VM as an OVF template. This functionality could be used in a number of ways:
 - Creating a template that could be transported between multiple vCenter Server instances
 - Transporting a VM from one vSphere installation to another vSphere installation
 - Transporting a VM to or from a different hypervisor that supports the OVF standard
 - Allowing a software vendor to package its product as a VM and easily distribute it to customers



3.Using OVF Templates

3.2Exporting a VM as an OVF Template



- Demo with vSphere Web Client
 - Exporting a VM as an OVF Template



3.Using OVF Templates

3.3Examining OVF Templates

- Three files make up the OVF template you exported out of vCenter Server:
 - The name of the manifest file ends in .mf, and the file contains SHA-1 digests of the other two files. This allows vCenter Server (and other applications that support the OVF specification) to verify the integrity of the OVF template by computing the SHA-1 digests of the other files in the package and comparing them against the SHA-1 digests in the manifest file. If the digests match, then the contents of the OVF template have not been modified.



3.Using OVF Templates

3.3Examining OVF Templates

- Three files make up the OVF template you exported out of vCenter Server:
 - The OVF descriptor is an XML document with a filename ending in .ovf and containing information about the OVF template such as product details, virtual hardware, requirements, licensing, a full list of file references, and a description of the contents of the OVF template.
 - The virtual hard disk file has a filename ending in .vmdk. The OVF specification supports multiple virtual hard disk formats, not just the VMDK files used by VMware vSphere, but obviously vCenter Server and VMware ESXi only natively support virtual hard disks in the VMDK format. Depending on the OVF template, it may contain multiple VMDK files, all of which would need to be referenced in the OVF descriptor file.



4.Using Content Libraries

- Content Libraries are a way of storing VMware templates, OVF templates, ISO/FLP media files, or any file that you may want cataloged separate from your deployed VMs. They can be synchronized between vCenter Servers to allow a “publish once, consume elsewhere” scenario.
- You can even subscribe to a Content Library that you might not own, such as a public Content Library from your favorite Linux distribution or maybe a private library from your virtual firewall vendor.



4.Using Content Libraries



- Demo with vSphere Web Client
 - Content Library Data and Storage
 - Content Library Synchronization
 - Creating and Publishing a Content Library
 - Subscribing to a Content Library
 - Operating Content Libraries
 - Importing Files to a Content Library
 - Importing VM Templates to a Content Library
 - Synchronizing Content Libraries



5. Working with vApps

- With vApps, vSphere administrators can combine multiple VMs into a single unit.
 - Increasingly, enterprise applications are no longer constrained to a single VM. Instead, they may have components spread across multiple VMs.
 - For example, a typical multitier application might have one or more front-end web servers, an application server, and a backend database server.
 - Although each of these servers is a discrete VM and could be managed as such, they are also part of a larger application that is servicing the organization.
 - Combining these different VMs into a vApp allows the vSphere administrator to manage the different VMs as a single unit.



5. Working with vApps



- Demo with vSphere Web Client
 - Creating a vApp
 - Editing a vApp
 - Editing a vApp's Resource Allocation Settings
 - Editing a vApp's IP Allocation Scheme
 - Editing a vApp's Authoring Settings
 - Editing a vApp's Power Settings
 - Changing a vApp's Power State
 - Cloning a vApp



