Mike Nelson – SSV article

Deploying Servers in Virtualization Using MDT

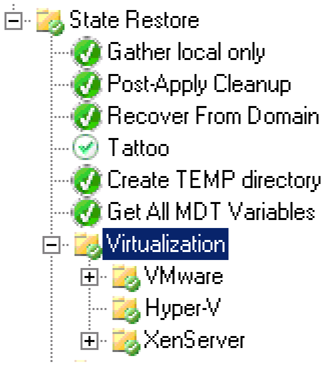
Automating Operating System deployments in both the physical and virtual datacenter world is a passion of mine and I want to take a deeper look at how to refine this process in the server virtualization space. Creating an automated OS deployment solution involves planning, time, and resources, and I won't be covering all of those aspects in this article, but rather I'll be focusing on some specifics of using the Microsoft Deployment Toolkit (MDT) and some tips to get your deployments off the ground.

For automating server (and desktops too!) OS deployments, either virtual or physical, I use the MDT from Microsoft. I use it as a standalone product, but it could be paired with a licensed Systems Center Configuration Manager (SCCM) to add more deployment and management features, such as Updates management and package distribution, to what it has already in the product. The MDT is a **free** Solution Accelerator package that is not limited in functionality and is fully supported by Microsoft. The latest version released is 2012 (Update 1 is now in Beta) and it has a couple of improvements over the previous version when it comes to working with virtualized workloads. I'll highlight those as we go along. If you do not already use the MDT or SCCM, I suggest you head over to the MDT site at Microsoft to get all of the information you need. I also have a listing of MDT resources available at my tech website.

One of the cool things about using the MDT to deploy virtual servers is that it "knows" when you are deploying to a virtualized environment. By using a script, called ZTIGather.wsf, to collect all the information it can about the physical or virtual hardware that is being deployed, it can identify virtualized processor types and sets a variable called "IsHypervisorRunning" to either "True" or "False". If it is "True", it will set a property variable called "IsVM" and set that to "True". And finally, it will set the platform variables. In the 2012 version, Microsoft has expanded the MDT to be able to detect these hypervisors - Microsoft's Hyper-V, VMware's vSphere, Citrix's Xenserver, or Oracle's VirtualBox. The following are the variables to use when setting commands in your task sequences:

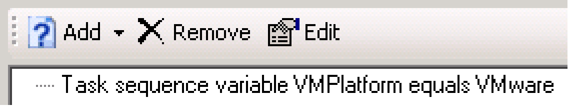
Microsoft Hyper-V – VMPlatform = Hyper-V
VMware vSphere – VMPlatform = VMware
Citrix Xenserver – VMPlatform = Xen
Oracle's VirtualBox – VMPlatform = VirtualBox

As an example, by incorporating these as a command sequence to install the proper Guest Tools (in this example VMware Tools) inside of the OS, you can create a Virtualization sequence folder like the one pictured below that will launch if the "IsVM" variable is equal to "True", and then evaluate the VMPlatform variable to see if it equals "VMware", and if it does, execute a task of running the silent install of the VMware Tools in the deployment -



- State Restore
  - Gather local only
  - Post-Apply Cleanup
  - Recover From Domain
  - Tattoo
  - Create TEMP directory
  - Get All MDT Variables
  - Virtualization
    - VMware
    - Hyper-V
    - XenServer

Figure 1 - Virtualization Folder in State Restore Sequence



Task sequence variable VMPlatform equals VMware

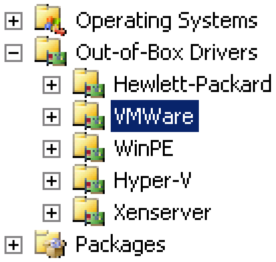
Figure 2 The VMPlatform Variable setting for VMware

As you can see from the screenshot, you can add all four VMPlatform variables of the hypervisors to your task sequence so that you can deploy on any platform that may be added to your datacenters and makes the MDT more scalable and flexible for future hypervisor technologies.

There are also many other variables that were added to the 2012 version that you can use in defining your task sequences and commands. Here is a brief listing of just some of what ZTIGather collects. You can always see what exactly is collected by ZTIGather by looking at its log file, ZTIGather.log, located in the %temp% directory.

```
SupportsHyperVRole is now = False
VMName is now = MACHINENAME1
SupportsVT is now = False
Supports64Bit is now = True
VMHost is now = HOSTNAME1.DOMAIN.COM
SupportsNX is now = True
```

Drivers are another issue to take into consideration when getting your deployment ready for creating virtual servers. All of the hypervisors have their own set of drivers that are needed in order to start the deployment of the OS with WinPE (part of the initial boot deployment process), and during the deployment for any hardware that is detected after the OS is installed. You will need to get familiar with identifying the different toolsets supplied by the hypervisor vendors and how to extract those drivers from their distributions and add them to the Out-Of-Box Drivers Store in MDT. By importing them into this store and organizing them by platform, you will save yourself a lot of time and frustration in trying to figure out what drivers are needed for what hypervisor platform.



- Operating Systems
- Out-of-Box Drivers
  - Hewlett-Packard
  - VMWare
  - WinPE
  - Hyper-V
  - Xenserver
- Packages

Figure 3 - Example Out-Of-Box Driver Store

I also would highly recommend enabling Monitoring in your MDT deployments, especially for virtual server installations. The reason for this is that you are able to deploy several (and by that I mean as many as your network will handle) deployment, across multiple hypervisor platforms, and even across multiple sites, all at one time! This can get a little difficult to manage and keep an eye on all of them, so the 2012 version includes a way to monitor their progress and report back on several aspects of all the deployments in a single console. When I was deploying even just 3 servers at one time this came in very handy as I did not have to have all those console windows open to see what was going on with the progress. Enabling monitoring is as easy as specifying in the properties of the deployment share that you want monitoring enabled, and also making sure that the Event and Data firewall ports are open on the MDT server. Michael Niehaus (the guy behind the MDT) also wrote a good feature article on it here.

I find that working with the MDT and SCCM is really quite fun actually. You can do so many different things, do it your own way or "borrow" someone else's way of doing it that they have published, and make a solid deployment solution for your virtual (and physical too!) infrastructure. Oh, and did I mention it's **FREE**?