Introduction to Azure Virtual Machines

Hope you have a good foundational understanding of Azure IaaS and PaaS. If not, you should through the **Azure Essentials** and **Azure Essentials Continuum** courses before going any further.

In this course, you will learn:

* Planning for Azure Virtual Machine (VM)
* VM creation using Custom Image, template and PowerShell
* Creation of Azure Linux VM
* Configuration of IP address, AV set, and VM Scale set
* Managing and Monitoring VM

***Note:****This course has been curated using the materials/resources received through our partnership with Microsoft. The course content is referred from the official*[*Microsoft*](https://openedx.microsoft.com/)*site.*

On-Premises vs Azure Virtual Machines

In Microsoft Data-centers, **Azure Virtual Machines (VMs)** are hosted on **Windows Server 2012 R2 Hyper-V servers**. But these VMs are different from On-Premises VMs in the following aspects:

* Console access is not given for Azure VMs, and it can be accessed only through RDP.
* Azure supports Generation 1 VMs only.
* Azure VMs do not support **VHDX** format.
* Azure VMs do not support OS upgrade.
* Azure VMs depend on the VM size and support more than 1 NICs.

Deployment Scenarios

Typical deployment scenarios of VMs are listed below,

* Create **Test and Dev Environments**
* **Extending your data center to the cloud**
* **Hosting certain applications in the cloud** that can leverage benefits of the cloud. **Example - Seasonal Applications**.
* **Installing Recovery Site** using IaaS-based DRS approach that provides significant costs savings.
* **High-performance computing** to solve complex problems involving millions of variables or calculations such as an earthquake.
* **Big data analysis** that involves processing and mining massive datasets.

##### Planning Considerations

While planning for the virtual machine deployment, you have to consider the following,

1. **Suitable and Unsuitable Workloads**
2. **Supported and Unsupported Server Roles**
3. **Supported and Unsupported Server Features**

These topics will be discussed in the next few cards.

1. Suitable Workloads

Azure VMs can be used for the following Workloads.

* **Highly available service workloads**. E.g., Online Stores.
* **Unpredictable Spikes** - E.g., News Channels.
* **Periodic workloads** - E.g., Retail sales spurt during Holiday Season.
* **Steady workloads** - E.g., Extend or offload existing infrastructure to the cloud.

##### Unsuitable Workloads

When Planning Azure VMs, it is also important to understand that not every application or service is a suitable fit for the cloud.

**Examples**

* **Low volume or limited growth workloads** - such services or applications can be run on commodity hardware on-premise and will be less expensive than in the cloud.
* **Regulated environment workloads** - Certain data is regulated by an organization or the local government. Such restricted and confidential data must be kept on-premises

2. Unsupported Server Roles

**Most of the Windows Roles can be enabled on Azure VMs, but few Roles are not supported in Azure VM**.

* Dynamic Host Configuration Protocol Server
* Hyper-V (Hyper-V role is supported in Azure Ev3 and Dv3 series VMs only)
* Rights Management Services
* Windows Deployment Services

3. Unsupported Server Features

**The following significant features are not supported**.

* BitLocker Drive Encryption (on the operating system hard disk, may be used on data disks)
* Internet Storage Name Server
* Multipath I/O
* Network Load Balancing
* Peer Name Resolution Protocol
* RRAS
* DirectAccess
* SNMP Services
* Storage Manager for SANs
* Windows Internet Name Service
* Wireless LAN Service

##### Azure Cost Optimization Tools

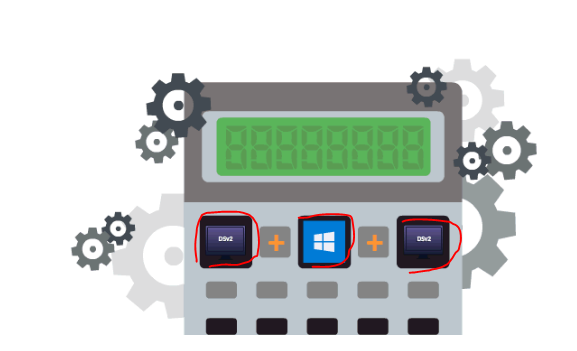
Now you know the consideration, the next major planning factors with any cloud-based service are:

* **Availability of resources**
* **Cost optimization**.

To help with estimating the **potential costs** and **to achieve the cost optimization** following tools are used:

* [**Pricing Calculator**](https://azure.microsoft.com/en-us/pricing/calculator/) tool enables you to estimate the cost of different workloads and services in Microsoft Azure.
* [**TCO Calculator**](https://www.tco.microsoft.com/) estimates the cost savings that can be realized by migrating the application workloads to Microsoft Azure.

##### Sizing and Configuration



[Pricing Calculator](https://azure.microsoft.com/en-in/pricing/calculator/) can be used to estimate the costs for Azure VMs.

**VM size and Configuration decides the Cost of the VM**

**VM sizing** is based on,

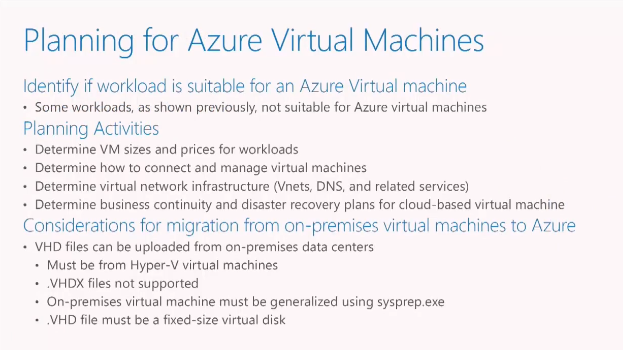
* **Compute**: Capacity required
* **Storage**: Size, location, and configuration
* **Disk**: Size, persistence and caching

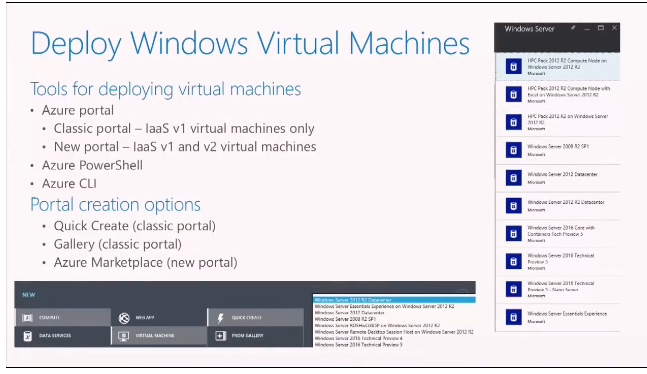
**VM configuration** deals with,

* **Operating System**: Windows Server 2012 R2, Windows Server 2016
* **IP Address allocation**: Static or Dynamic IP addresses
* **Availability**: Uptime requirements, geo-distribution, service level agreements, and accessibility.
* **Scale set**: Type of scaling and threshold

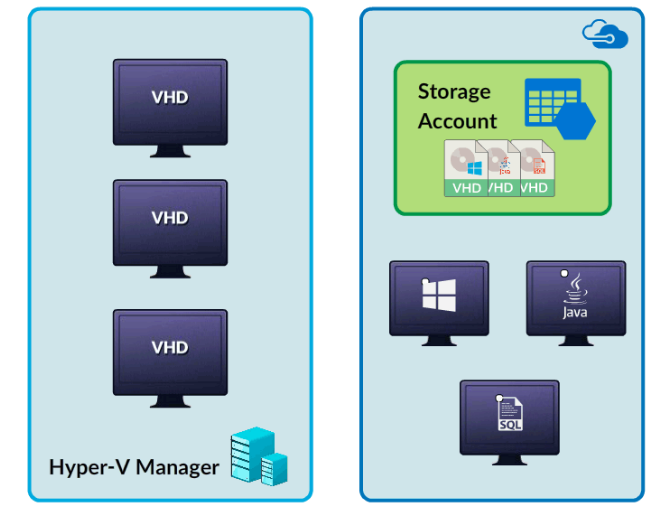
##### More on Planning for VMs

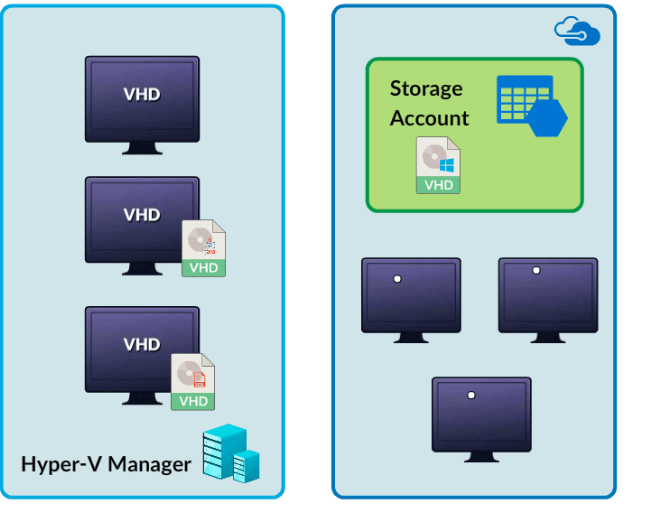
This video is the final note on the planning for VMs before we move on to Creation of Virtual Machines.





##### Creating VM Using Custom Image





In the course Azure Essentials, you learned how to create VMs using the OS images available in the Azure Market Place.

Now we will create VMs using a Custom image. Images can be created for **Generation 1 VMs** that are in the VHD file format.

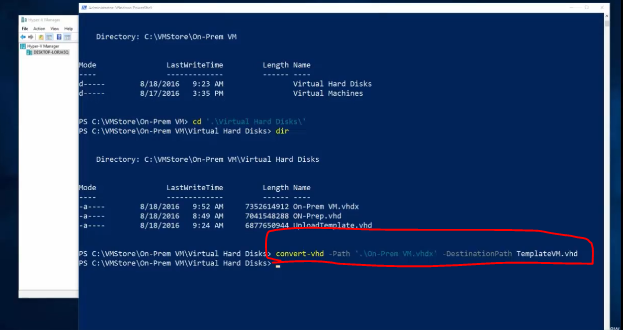
Steps to create VM using Custom Image:

**On-Premise**

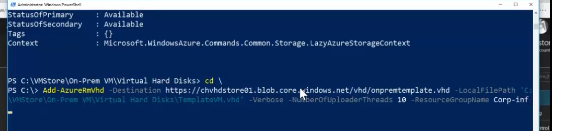
* Prepare the VM. Make sure the VM has all the roles and features installed.
* Run sysprep to prepare the machine.
* Prepare the VM VHD.

**Azure On-Premise**

* Create the Storage Container.
* Upload the VHD.
* Create a VM using the uploaded VHD.



Adding vhd file to azure storage account by using powershell.



Login to azure account from powershell –

**Login-AzureRmAccount - RM is Resource manager(ARM)**



Get the storage details. – Get-AzureStorageAccount



##### VM Creation Using Template

##### Getting Started with VM Creation

Now that you understand how to plan for VMs, next we will learn various ways of **Provisioning VMs**.

**Tool for deploying VMs**:

* Azure Portal
  + Classic portal: **V1 VMs only**
  + New portal: **Both V1 and V2 VMs**
* Azure PowerShell
* Azure CLI
* ARM Template

**Versions of VM (V1 and V2) are different from Generations of VM such as G1 and G2**.

Azure supports G1 VMs only

##### OS Image Sources

**Azure Marketplace**

* Contains recent versions for Windows and Linux distributions

**VM Depot**

* Community managed Repository of Linux and FreeBSD VM images

**Custom Images**

* Enterprise OS images with Applications that must be captured from VM in the Organization and uploaded in Azure to create further VMs

##### ARM Templates

**Resource Manager templates are JSON files that define the resources to be deployed for an Azure solution**.

**JSON editor** or **Visual Studio Code** can be used to create templates.

**Prerequisites to create a template**:

* [Visual Studio Code](https://code.visualstudio.com/).
* Azure subscription.

##### Features and Benefits of ARM Templates

ARM Templates allow you to **deploy identical environments** to test staging and production, ensuring high reliability.

* Templates can be stored in the Azure Subscription and reused.
* An entire Resource group can be exported as a Template.
* ARM Templates can create resources in parallel.
* ARM Templates can define dependencies among resources.
* ARM Templates can create custom policies to control actions or operations that can be performed on resources.

Importantly, ARM templates help to reduce human error when deploying complex environments.

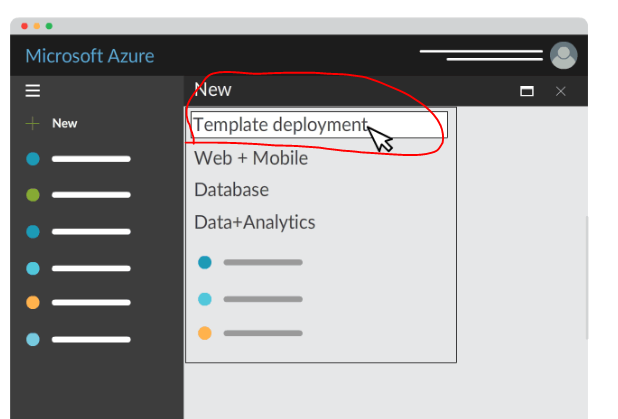
##### Features and Benefits of ARM Templates

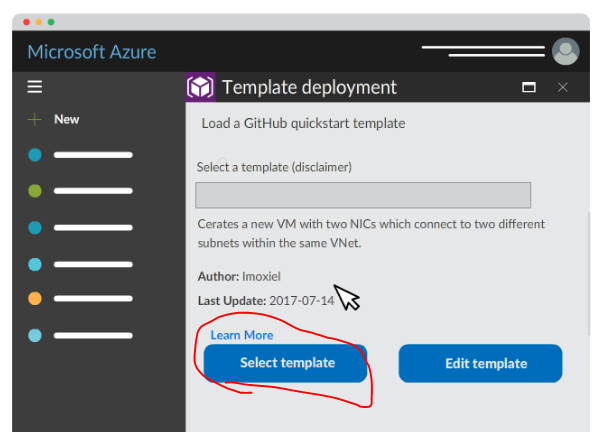
ARM Templates allow you to **deploy identical environments** to test staging and production, ensuring high reliability.

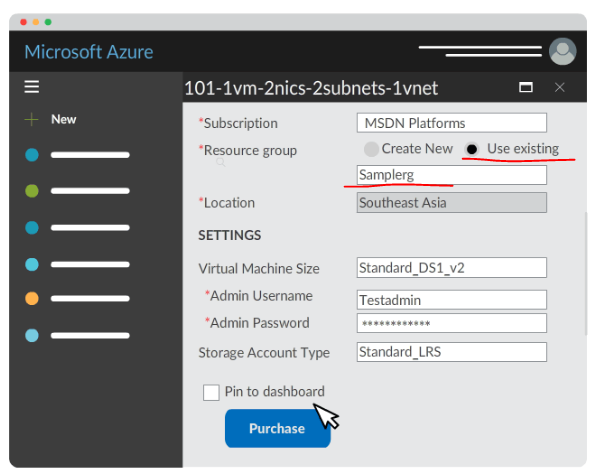
* Templates can be stored in the Azure Subscription and reused.
* An entire Resource group can be exported as a Template.
* ARM Templates can create resources in parallel.
* ARM Templates can define dependencies among resources.
* ARM Templates can create custom policies to control actions or operations that can be performed on resources.

Importantly, ARM templates help to reduce human error when deploying complex environments.

##### Deploying a VM Using ARM Template







**ARM template is another way to create one or more Azure VMs quickly**.

**Steps**:

* **Create the Template**: You can create your own template or you can get it from Azure Quickstarts Templates.
* **Create the Parameter File**: To specify values for the resource parameters that were defined in the template or parameters files.
* **Create the Resource Group**: All resources, like virtual machines, must be part of a resource group.
* **Deploy the Template**: Once the template and resource files are ready, use the **New-AzureRmResourceGroupDeployment** command to implement the resource.

**The template file can be created with a simple text editor and the file type is JSON**.

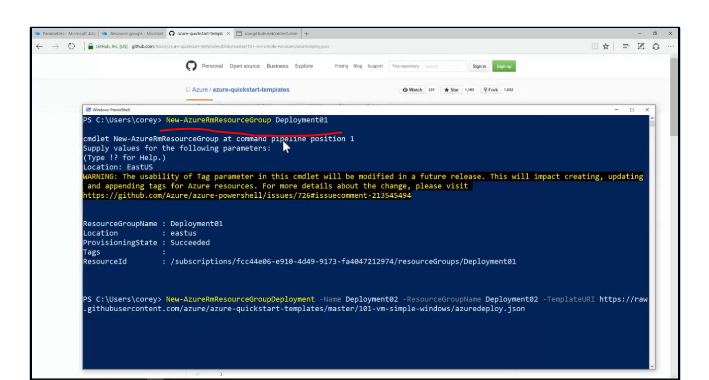
##### Template Based Deployment Using PowerShell

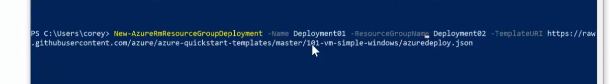
Watch this video to understand, how to use **GitHub Quick Start Templates** for deploying a VM using **PowerShell**.

Templates

<https://azure.microsoft.com/en-us/resources/templates/>

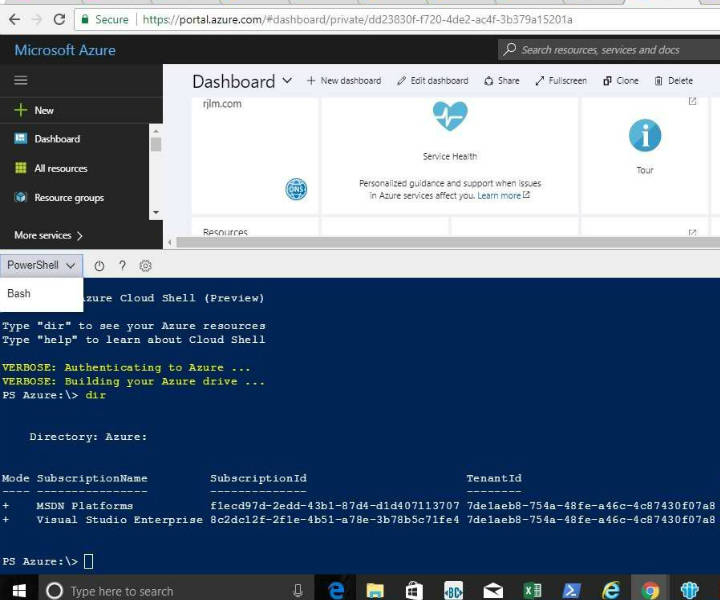
Creating deployment grp





##### VM Creation Using PowerShell

##### Azure Cloud Shell



Another popular way of managing VMs is **using PowerShell or CLI interface**. For this, you would need Azure Cloud Shell.

**Azure Cloud Shell is an interactive, browser-accessible shell for managing Azure resources**.

**Features of Azure Cloud Shell:**

* It is a free shell that can be **run directly on the Azure portal**.
* It has common Azure tools **preinstalled and configured** to use with your account.

**Click the Cloud Shell button** on the menu in the upper-right of the Azure portal, and you will able to launch both **PowerShell and CLI interface**.

It gives you the flexibility of choosing the shell experience that best suits the way you work.

##### Knowing Cloud Shell

* It runs on a temporary machine provided on a per-session, per-user basis.
* It times out after 20 minutes without interactive activity.
* It can only be accessed with a file share attached.
* It uses the same file share for both Bash and PowerShell.
* It is assigned to a one machine per user account basis.
* Permissions are set as a regular Linux user (Bash).

##### Authentication and Pricing

**Automatic authentication**

* Cloud Shell securely and automatically authenticates on each session for instant access to your resources through the Azure CLI 2.0.

**Pricing**

* The machine hosting Cloud Shell is free, with a prerequisite of a mounted Azure file share.
* Regular storage costs apply.

**Use Cloud shell to create and manage the VMs**.

##### Steps to Create VM using PS

1. Create Azure Connection
2. Create resource group
3. Create networking resources such as subnet, Vnet, NIC, Public IP Address.
4. Define a credential object
5. Create a VM configuration
6. Create virtual machine

PowerShell commands for above steps are uploaded in the Azure storage can be accessed from:

**Azure Portal - Storage - frescostepguide - file - Azure VM - VM\_Creation\_PS.pdf**

##### VM Creation using PS

Using listed PS commands in the following cards you will be able to create a VM with configuration as below,

* **ResourceGroup** - myResourceGroup
* **Location** - EastUS
* **Subnet** - mySubnet
* **AddressPrefix**- 192.168.1.0/24
* **vnet** - MYvNET
* **AddressPrefix** - 192.168.0.0/16
* **VMSize** - Standard\_DS2
* **ComputerName** - myVM
* **OS** - Windows Server 2016 datacenter

##### Demo: VM Installation using PowerShell

Watch this video to learn how to create a basic Windows VM using Azure PowerShell cmdlets.

##### Linux on Azure

**Microsoft Endorsed some versions of Linux** in the Azure Marketplace in which,

* The **Linux Integration Services (LIS) drivers for Hyper-V** and **Azure** are kernel modules that are contributed directly from MS to the upstream Linux kernel.
* The **Azure Linux Agent** is already pre-installed and the source code can be found on GitHub.

**For Non-Endorsed Versions of Linux**

* You can bring your own Linux image by Creating and uploading a virtual hard disk that contains the Linux OS.

##### Endorsed distributions

Azure SLAs applies only to the Endorsed versions

* CentOS 6.3+
* CoreOS 494.4.0+
* Debian 7.9+, 8.2+
* Oracle Linux 6.4+, 7.0+
* RHEL 6.7+, 7.1+
* SLES 11, SLES 12+, SLES for SAP11.3+
* Open SUSE 13.2+
* Ubuntu 12.04, 14.04, 16.04

Planning of Linux VM

* 

Planning and Sizing of VM is same as Windows VMs.

* Azure Marketplace includes a large number of Linux virtual machines and the ones in the above illustration are just a few to mention.

##### Creating Linux VM

This video explains the creation of Azure Linux VM using ARM Portal and Azure Cloud Shell.

Azure vm list – it will display the list of vm which are under my subscription.

Once VM created,Please do change the password of linux vm to login.



##### Accessing of Linux VMs

Once the Linux VMs are created, it is required to access the VM.

Here are the few ways to access the VMs,

* **Azure Cloud Shell (CLI)**
* **Bash Shell from Windows 10**
* **Putty software**

Next few cards explains how to manage Linux VM from Windows 10 Environment

Managing of Linux using Windows

A common problem encountered is how to easily manage Linux virtual machines from Windows environments.

**Windows 10 now makes it possible to run Linux Bash natively.**

**Bash on Windows allows you to:**

* **Run common command-line utilities** such as grep, sed, and awk.
* Use the Linux-compatible filesystem & heirarchy and **access fixed Windows storage**mounted under "/mnt/..."
* **Install additional Linux tools** using apt
* **Invoke Windows applications** from within Bash
* **Invoke Linux applications** from within Windows
* **Run Bash shell scripts and Linux command-line apps**, including:
  + Tools: vim, emacs, tmux
  + Languages: Javascript/node.js, Ruby, Python, C/C++, C# & F#, Rust, Go
  + Services: sshd, MySQL, Apache, lighttpd

Installing Bash on Windows10

**Installing Bash on Windows 10:**

* Turn on Developer mode via Settings > Update and Security > For Developers.
* Enable Bash in Windows by adding the Windows feature "Windows Subsystem for Linux (beta)".
* Restart the Windows 10 workstation.

**It can be enabled using PowerShell as:**

Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux

**Accessing Bash on Windows 10**:

* Launch the command line and run the command Bash, to enter the Linux subsystem. (or)
* From the Start Menu select Bash on Ubuntu on Windows.

##### Working with Bash Shell

This video explains how to connect the VM using Bash Shell. Also it explains about managing resources using Bash Shell\Azure CLI.