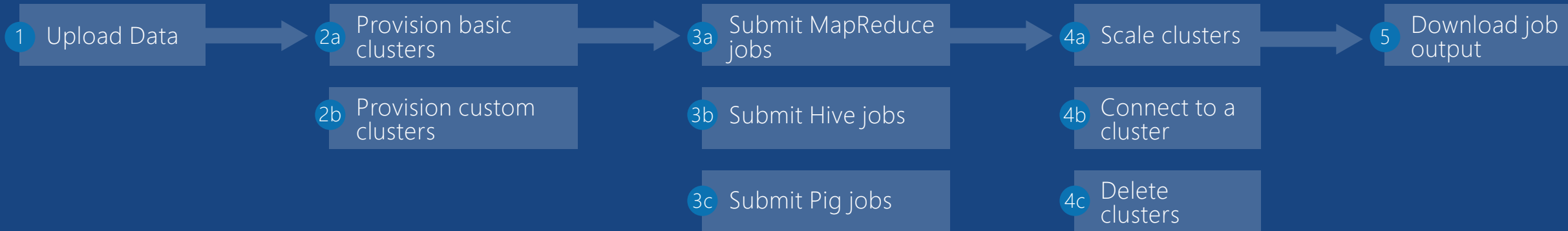




# PowerShell cmdlet cheat sheet: Service Management Model

Microsoft Azure  
HDInsight

Download this PowerShell cmdlet cheat sheet for common Hadoop tasks



## Variables

```
$storageAccName = "<Default storage account>"
$containerName = "<container name in the storage account>"
$clusterName = "<HDInsight cluster name>"
$location = "<Data center location for the cluster>"
$clusterNodes = "<Number of cluster nodes>"
$sshPublicKey = "<SSH public key>"
```

## Connect to Azure and select a subscription

```
# Connect to an Azure account
Add-AzureAccount

# List the Azure subscriptions for the account
Get-AzureSubscription

# Select an Azure subscription
Select-AzureSubscription -SubscriptionName <subscription name>
```

## 1 Upload data to a cluster storage

```
$fileName = "<LocalFileName>"
$blobName = "<BlobName>"

# Get the storage account key
$storageAccKey = Get-AzureStorageKey $storageAccName | %{$_.Primary}

# Create the storage context object
$destContext = New-AzureStorageContext -StorageAccountName $storageAccName -StorageAccountKey $storageAccKey

# Copy the file from local workstation to the Blob container
Set-AzureStorageBlobContent -File $fileName -Container $containerName -Blob $blobName -context $destContext
```

## 2a Provision basic Linux clusters

```
# Get the Storage account key
$storageAccKey = Get-AzureStorageKey $storageAccName | %{$_.Primary}

$sshCredentials = Get-Credential

# Create a new HDInsight cluster
New-AzureHDInsightCluster -Name $clusterName -Location $location -DefaultStorageAccountName "$storageAccName.blob.core.windows.net" -DefaultStorageAccountKey $storageAccKey -DefaultStorageContainerName $containerName -Version <string> -OSType Linux -ClusterSizeInNodes $clusterNodes -SshCredential $sshCredentials -SshPublicKey $sshPublicKey
```

## 2b Provision custom Linux clusters (e.g. with specific head node size)

```
$nodeVMSize = "<Node size>"
$sshCredentials = Get-Credential

# Get the Storage account key
$storageAccKey = Get-AzureStorageKey $storageAccName | %{ $_.Primary }

# Create a new HDInsight cluster
New-AzureHDInsightCluster -Name $clusterName -Location $location -DefaultStorageAccountName "$storageAccName.blob.core.windows.net" -DefaultStorageAccountKey $storageAccKey
-DefaultStorageContainerName $containerName -ClusterSizeInNodes $clusterNodes -HeadNodeVMSize $nodeVMSize -Version <string> -OSType Linux -SshCredential $sshCredentials
-SshPublicKey $sshPublicKey
```

## 3a Submit MapReduce jobs

```
$jobDefinition = New-AzureHDInsightMapReduceJobDefinition -JarFile "<fully qualified name of the jar file>" -ClassName "<name of the job class in the jar file>"
-Arguments "<string arg1>", ..., "<string arg n>"

$jobName = Start-AzureHDInsightJob -Cluster $clusterName -JobDefinition $jobDefinition

Wait-AzureHDInsightJob -Job $jobName -WaitTimeoutInSeconds <number of seconds>

Get-AzureHDInsightJobOutput -Cluster $clusterName -JobID $jobName.JobID -StandardOutput
```

## 3b Submit Hive jobs

```
$queryString = "<hive query>"

Use-AzureHDInsightCluster $clusterName
Invoke-Hive -Query $queryString
```

## 3c Submit Pig jobs

```
$queryString = @"<pig latin query>"

$pigJobDefinition = New-AzureHDInsightPigJobDefinition -Query $queryString

$pigJobName = Start-AzureHDInsightJob -Cluster $clusterName -JobDefinition $pigJobDefinition

Wait-AzureHDInsightJob -Job $pigJobName -WaitTimeoutInSeconds <number of seconds>

Get-AzureHDInsightJobOutput -Cluster $clusterName -JobID $pigJobName.JobID -StandardOutput
```

## 4a Show cluster details

```
# List all clusters in a subscription with their details
Get-AzureHDInsightCluster

# Show details for a specific cluster
Get-AzureHDInsightCluster -Name <cluster name>
```

## 4b Scale clusters (Windows only)

```
Set-AzureHDInsightClusterSize <new cluster size>
-Name <cluster name>
```

## 4c Delete clusters

```
Remove-AzureHDInsightCluster -Name <cluster name>
```

## 5 Download job outputs

```
# Get the storage account key
$storageAccKey = Get-AzureStorageKey $storageAccName | %{ $_.Primary }

# Create the context object
$destContext = New-AzureStorageContext -StorageAccountName $storageAccName -StorageAccountKey $storageAccKey

# Use the -blob switch to filter blobs contained in the location you want
Get-AzureStorageBlob -Container $containerName -Blob <path to the blob> -Context $destContext | Get-AzureStorageBlobContent -Context $destContext
```