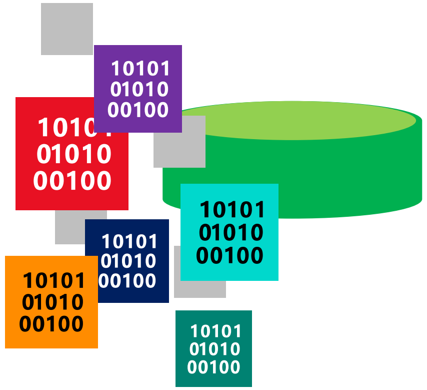
Introduction to Big Data and HDInsight

Microsoft

What is Big Data?

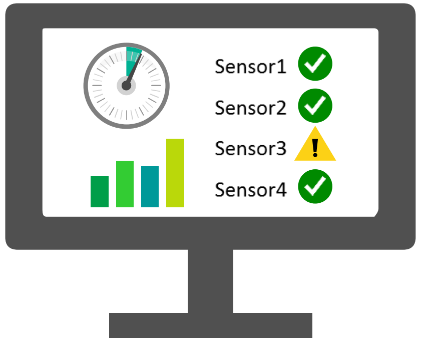
- Data that is too large or complex for analysis in traditional relational databases
- Typified by the “3 V’s”:
  - *Volume* – Huge amounts of data to process
  - *Variety* – A mixture of structured and unstructured data
  - *Velocity* – New data generated extremely frequently



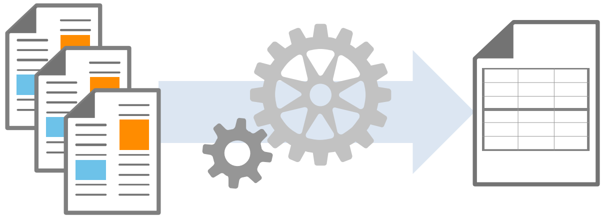
Web server click-streams

Social media sentiment analysis

Sensor and IoT Processing

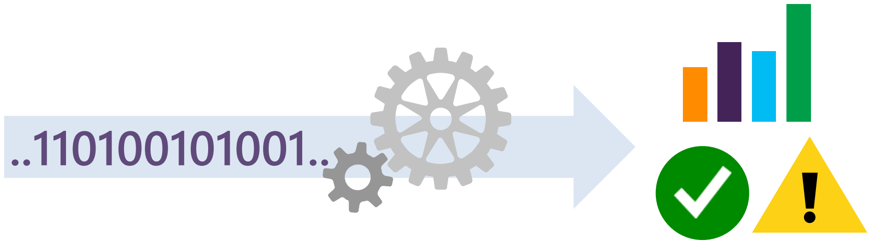


Batch Processing



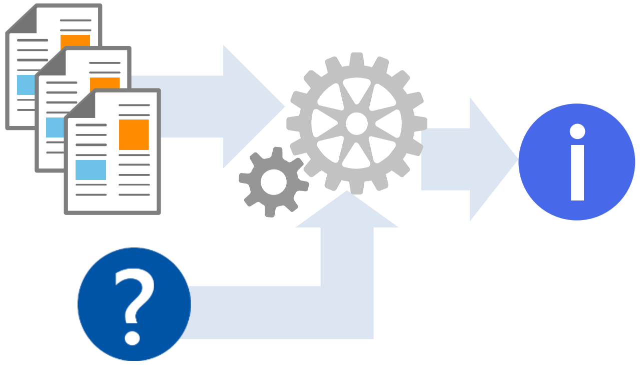
Filter, cleanse, and shape data for analysis

Real-Time Processing



Capture, filter, and aggregate streams of data for low-latency querying

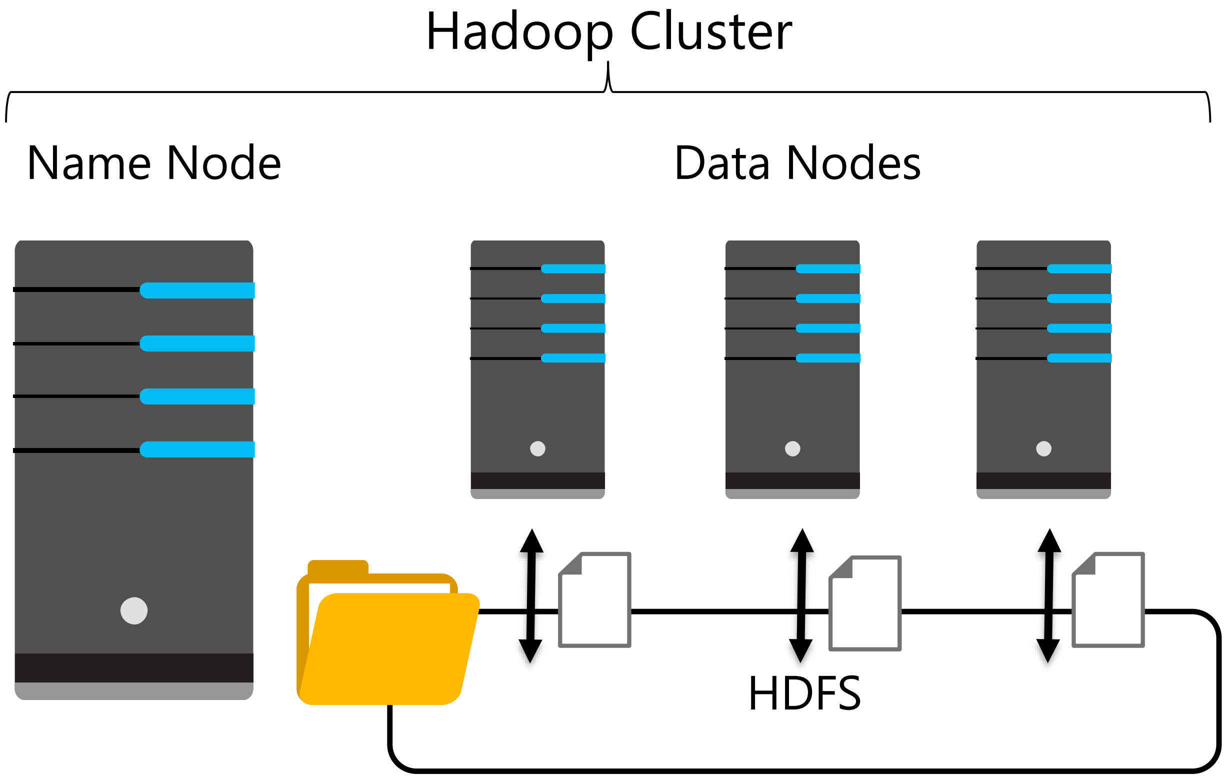
Predictive Analytics



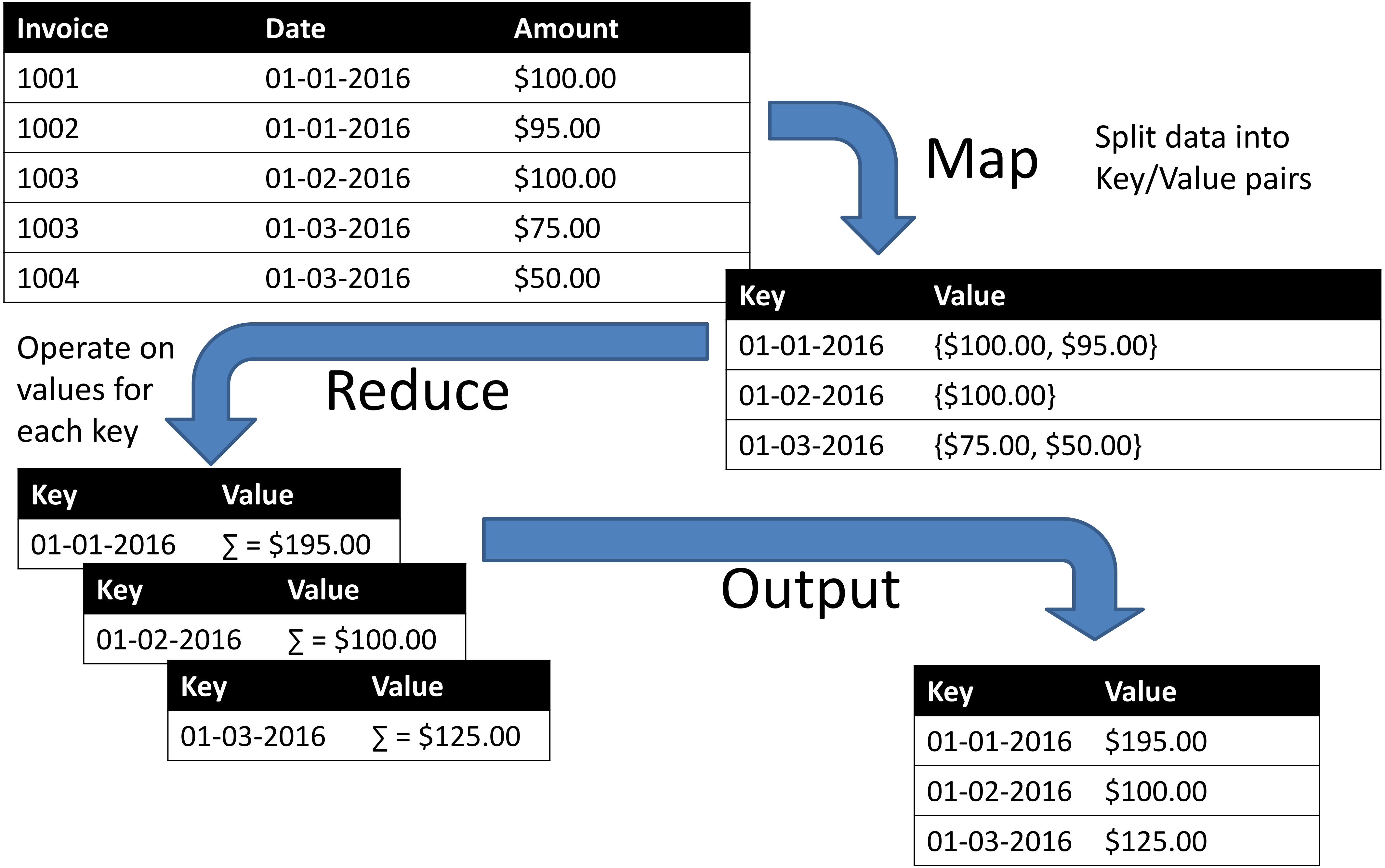
Apply statistical algorithms for classification, regression, clustering, and prediction

What is Hadoop?

- Hadoop
  - Open source distributed data processing cluster
  - Data processed in Hadoop Distributed File System (HDFS)
  - Resource Management is performed by YARN
- Related projects
  - Hive
  - Pig
  - Oozie
  - Sqoop
  - Others



What is MapReduce?



Invoice	Date	Amount
1001	01-01-2016	$100.00
1002	01-01-2016	$95.00
1003	01-02-2016	$100.00
1003	01-03-2016	$75.00
1004	01-03-2016	$50.00

Map

Split data into
Key/Value pairs

Key	Value
01-01-2016	{$100.00, $95.00}
01-02-2016	{$100.00}
01-03-2016	{$75.00, $50.00}

Operate on
values for
each key

Reduce

Key	Value
01-01-2016	∑ = $195.00

Key	Value
01-02-2016	∑ = $100.00

Key	Value
01-03-2016	∑ = $125.00

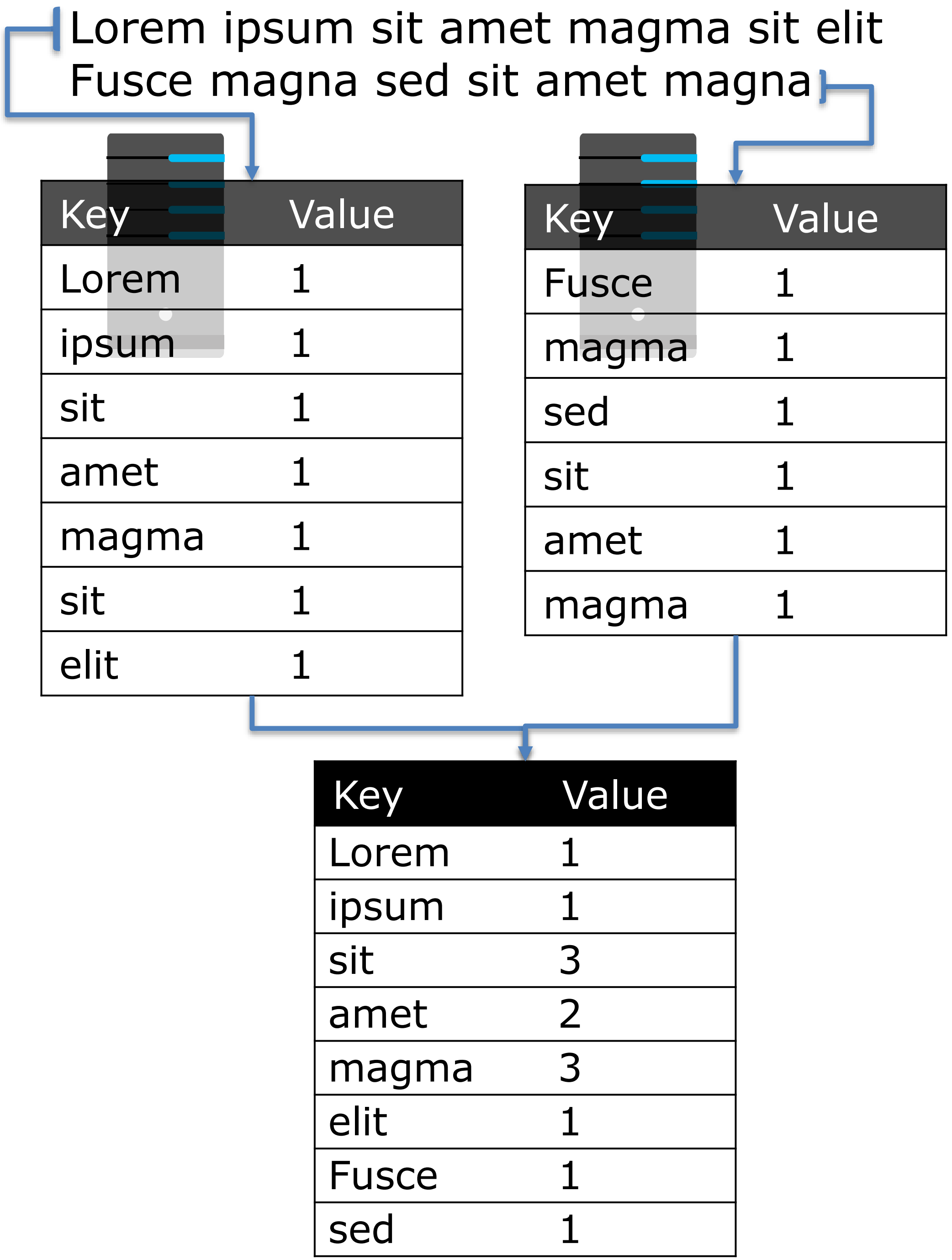
Output

Key	Value
01-01-2016	$195.00
01-02-2016	$100.00
01-03-2016	$125.00

Word Count

The “Hello World” of MapReduce

1. Source text is divided among data nodes
2. Map phase generates key/value pairs with words as keys and placeholder values of 1
3. Reduce phase aggregates values for each key by adding the values for each word



Lorem ipsum sit amet magma sit elit
Fusce magna sed sit amet magna

Key	Value
Lorem	1
ipsum	1
sit	1
amet	1
magma	1
sit	1
elit	1

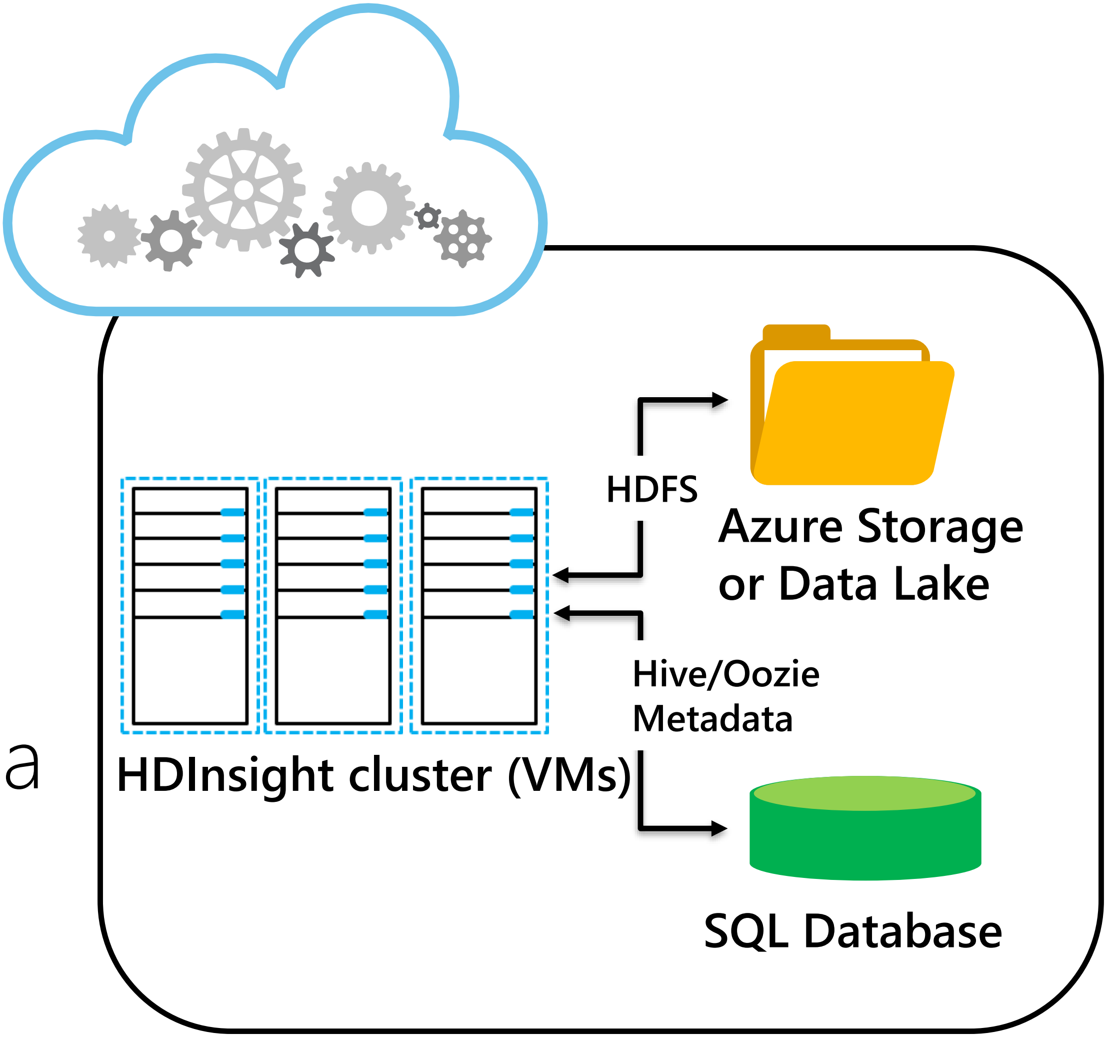
Key	Value
Fusce	1
magma	1
sed	1
sit	1
amet	1
magma	1

Key	Value
Lorem	1
ipsum	1
sit	3
amet	2
magma	3
elit	1
Fusce	1
sed	1

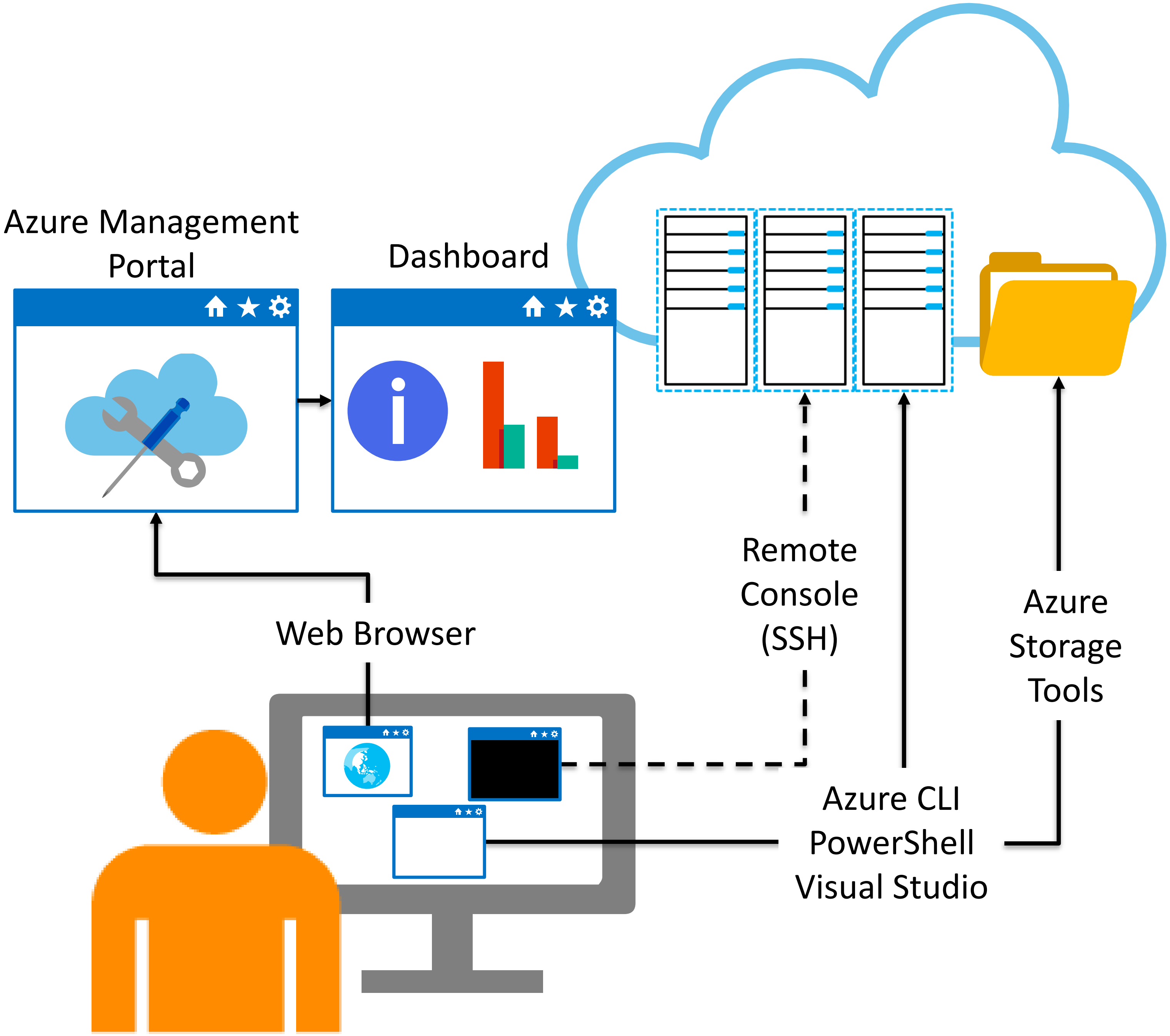
```
public static class Map extends Mapper<LongWritable, Text, Text, IntWritable> {
       private final static IntWritable one = new IntWritable(1);
       private Text word = new Text();
       public void map(LongWritable key, Text value, Context context) {
           String line = value.toString();
           StringTokenizer tokenizer = new StringTokenizer(line);
           while (tokenizer.hasMoreTokens()) {
               word.set(tokenizer.nextToken());
               context.write(word, one);
           }
       }
}


public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable> {
     public void reduce(Text key, Iterable<IntWritable> values, Context context){
       int sum = 0;
       for (IntWritable val : values) {
               sum += val.get();
       }
       context.write(key, new IntWritable(sum));
     }
}
```

What is HDInsight?

- Apache Hadoop on Azure
  - Hortonworks HDP on Azure VMs
- Azure Storage or Azure Data Lake provides the HDFS layer
- Azure SQL Database stores metadata



What client tools can I use?


Azure Management Portal
Dashboard
Web Browser
Remote Console (SSH)
Azure Storage Tools
Azure CLI
PowerShell
Visual Studio

File paths can be referenced using WASB(S) or native syntax

- wasb://*container*@*account*.blob.core.windows.net/data/logs/file.txt
- wasb:///data/logs/file.txt (default storage account and container)
  - or
- /data/logs/file.txt

File paths are **case-sensitive**

```
$>hdfs dfs ls /
```

HDFS shell commands

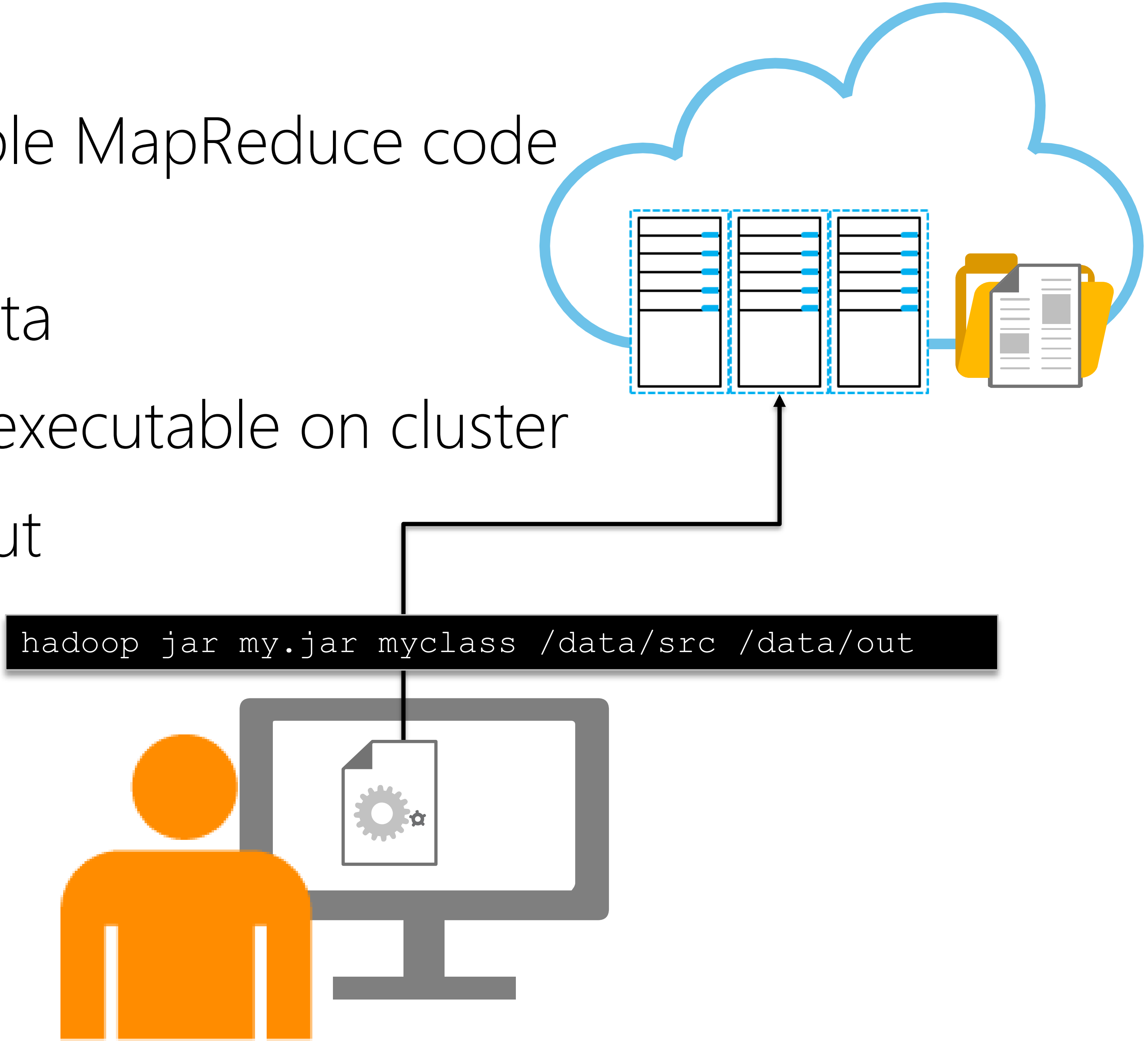
- **ls** (list)
- **cp** and **mv** (copy and move)
- **mkdir** (make directory)
- **rm** and **rm –r** (remove and remove recursive)
- **put** and **get** (transfer files between local file system and HDFS)
- **text**, **cat**, and **tail** (display contents of file)

How do I Run a MapReduce Job?

1. Compile executable MapReduce code

   Commonly a Java jar

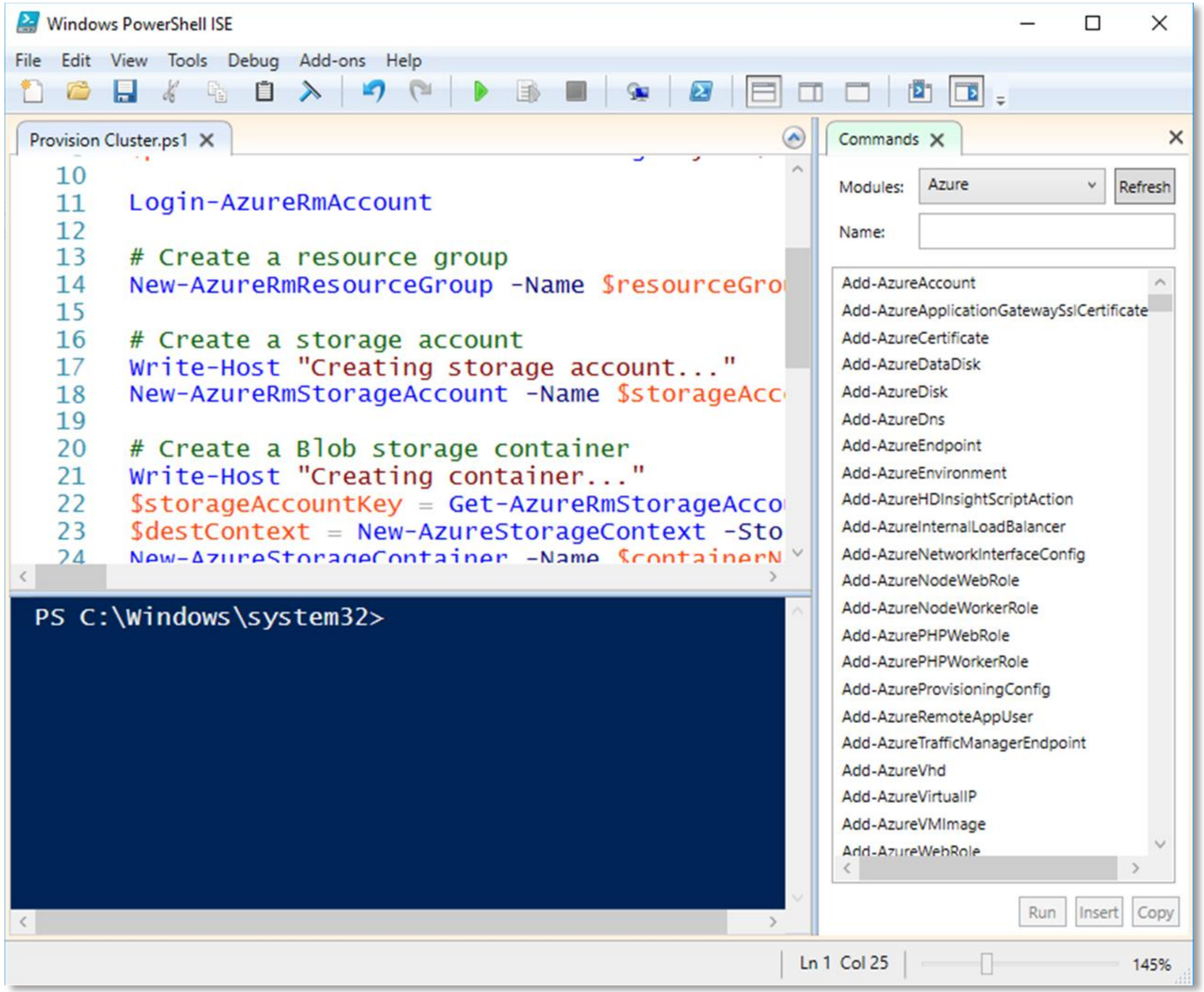
2. Upload source data
3. Run MapReduce executable on cluster
4. Retrieve job output



```
hadoop jar my.jar myclass /data/src /data/out
```

How do I use PowerShell with HDInsight?

- The Azure PowerShell module includes cmdlets to work with Azure services, including HDInsight
- Use PowerShell to:
  - Provision HDInsight clusters
  - Upload/download files
  - Submit jobs
  - Manage cluster resources



```

Login-AzureRmAccount

# Create a resource group
New-AzureRmResourceGroup -Name $resourceGro

# Create a storage account
Write-Host "Creating storage account..."
New-AzureRmStorageAccount -Name $storageAcc

# Create a Blob storage container
Write-Host "Creating container..."
$storageAccountKey = Get-AzureRmStorageAcco
$destContext = New-AzureStorageContext -Sto
New-AzureStorageContainer -Name $containerN
```

```
PS C:\Windows\system32>
```

Add-AzureAccount
Add-AzureApplicationGatewaySslCertificate
Add-AzureCertificate
Add-AzureDataDisk
Add-AzureDisk
Add-AzureDns
Add-AzureEndpoint
Add-AzureEnvironment
Add-AzureHDInsightScriptAction
Add-AzureInternalLoadBalancer
Add-AzureNetworkInterfaceConfig
Add-AzureNodeWebRole
Add-AzureNodeWorkerRole
Add-AzurePHPWebRole
Add-AzurePHPWorkerRole
Add-AzureProvisioningConfig
Add-AzureRemoteAppUser
Add-AzureTrafficManagerEndpoint
Add-AzureVhd
Add-AzureVirtualIP
Add-AzureVMImage
Add-AzureWebRole

Microsoft