MyShuttle Deployment Guide

Contents

[Prerequisites 2](#_Toc417910767)

[Step 1. Install prerequisites 2](#_Toc417910768)

[Step 2. Configure Azure credentials 3](#_Toc417910769)

[Step 3. Configure your environment 4](#_Toc417910770)

[Step 4. Register Azure Resources Providers 5](#_Toc417910771)

[Step 5. Create Azure resources for Apps demos 6](#_Toc417910772)

[Step 6. Create Azure resources for IoT and Big Data demos 9](#_Toc417910773)

[Step 7. Configure Apps 11](#_Toc417910774)

[Demo ASP.NET 11](#_Toc417910775)

[Demo Desktop App 11](#_Toc417910776)

[Demo Native Apps 11](#_Toc417910777)

[Demo Connected Systems 11](#_Toc417910778)

[Demo Cordova 12](#_Toc417910779)

[Demo IoT 13](#_Toc417910780)

[Demo Dashboard 15](#_Toc417910781)

[Step 8. Initialize the accelerometer aggregate Hive table 15](#_Toc417910782)

[Step 9. Create the Machine Learning experiments 15](#_Toc417910783)

# Prerequisites

Before starting you should have:

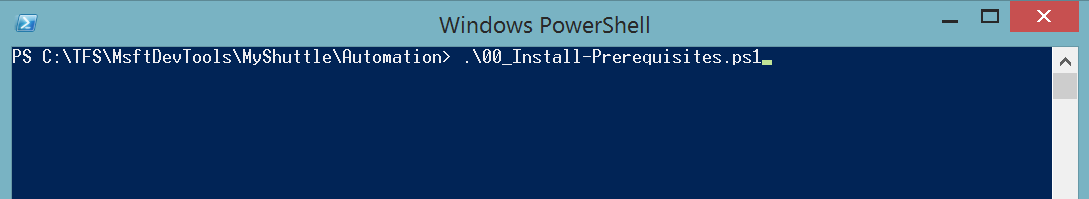
* Windows Azure Subscription.
  + [Get Microsoft Azure Trial](http://azure.microsoft.com/en-us/pricing/free-trial/)
* Visual Studio 2015.
  + [Get Visual Studio 2015 Preview](http://www.visualstudio.com/en-us/downloads/visual-studio-2015-downloads-vs.aspx)
* Bing Maps Key
  + [Getting a Bing Maps Key](http://msdn.microsoft.com/en-us/library/ff428642.aspx)
* Office 365 Account
  + [Get Office 365 Trial](https://products.office.com/en-us/try)
* Salesforce Account
  + [Get Safesfoce trial](https://www.salesforce.com/form/signup/freetrial-sales.jsp)

# Step 1. Install prerequisites

The installation also needs Windows Azure PowerShell and Microsoft Azure Cross Platform Command Line (azure-xplat-cli) but in this case you don't have to install this prerequisites by yourself, you can just run the script **01\_InstallPrerequisites.ps1** in a PowerShell console and the prerequisites will be installed automatically.

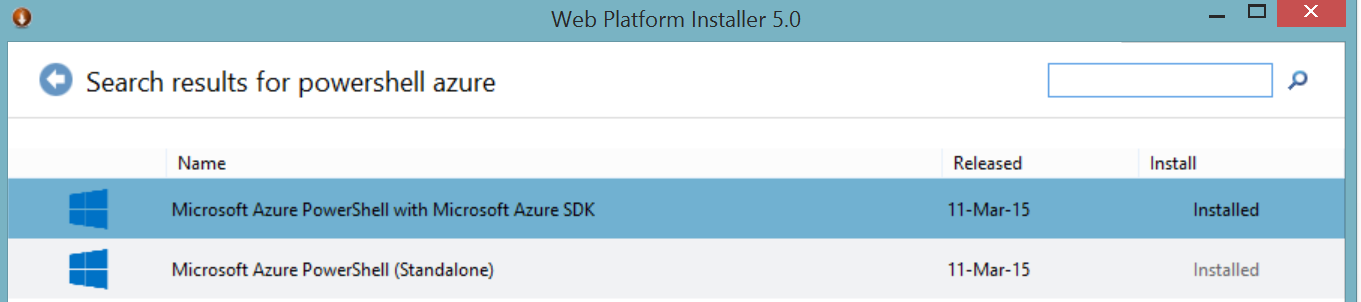
This script will install Windows Azure PowerShell and Microsoft Azure Cross Platform Command Line and all the dependencies using Chocolatey Package Manager.

* Open powershell cmd with ADMIN privileges.
* Before executing the PowerShell script, you’ll need to allow scripts executions in the policies of your machine or user. If you want to allow that at a local machine scope, run the following command in the PowerShell console:
  + **Set-ExecutionPolicy Unrestricted -Scope LocalMachine**
  + For more info about those policies, read the following: <http://technet.microsoft.com/library/hh847748.aspx>
* Launch .\**00\_InstallPrerequisites.ps1**



* Check you have the latest Azure PowerShell version using Web Platform installer.

Click on the “**Products**” tab and Install the **Microsoft Azure PowerShell and Microsoft Azure SDK** plus the **Microsoft Azure Cross-Platform Command Line Tools** if they are not already installed.



# Step 2. Configure Azure credentials

1. Open the “**Microsoft Azure PowerShell**” console.
2. Type the following command:  
     
   **Add-AzureAccount**
3. In the window, type the email address and password associated with your account.
4. Azure authenticates and saves the credential information, and then closes the window.
5. Your account could have several subscriptions related. Make sure which one you want to use and if required, select it with the **Select-AzureSubscription** command.
6. Once the right Azure subscription is selected, to download the publish settings for your account, use the following command:

azure account download

1. This will open your default browser and prompt you to sign in to the Azure Management Portal. After signing in, a .publishsettings file will be downloaded. Make note of where this file is saved.
2. NOTE: You can also download the PublishSettings file from the following URL: <https://manage.windowsazure.com/publishsettings/>
3. Next, import the .publishsettings file by running the following command, replacing [path to .publishsettings file] with the path to your .publishsettings file:

azure account import [path to .publishsettings file]

After importing your publish settings, you should delete the .publishsettings file, as it is no longer required by the Command-Line Tools and presents a security risk as it can be used to gain access to your subscription.

# Step 3. Configure your environment

1. Edit the **environment.xml** file and set the values of each parameter. You can change these values depending on your preferences and actual values for O365 and Salesforce.

IMPORTANT: Change the default demo resource names!

<env>

<azuresubscription name=" Windows Azure MSDN" />

<storageaccount name="myshuttletest" />

<resourcegroup name="myshuttletest" location="West US" />

<websiteaspnet name="myshuttletest" />

<websitemobileservice name="myshuttletestMS" />

<websitewebjob name="myshuttletestWJ" />

<sqlserver name="myshuttletest" login="myshuttleadmin" password="myshuttle1234!" dbname="MyShuttle" />

<sharepoint name="MyShuttleSharepointName" username="MyShuttleSharepointUsername" password="MyShuttleSharepointPassword" />

<salesforce consumerkey="SalesforceConsumerKey" username="SalesforceUsername" password="SalesforcePassword" consumersecret="SalesforceConsumerSecret" accountid="SalesforceAccountId" />

<eventhub namespace="myshuttletestns" name="myshuttletest" />

<hdinsight name="myshuttletesthd" clusternodes="1" username="admin" password="MyShuttle1#" />

<streamanalytics name="myshuttletest" location="Central US" />

<datafactory name="myshuttletest" location="West US" />

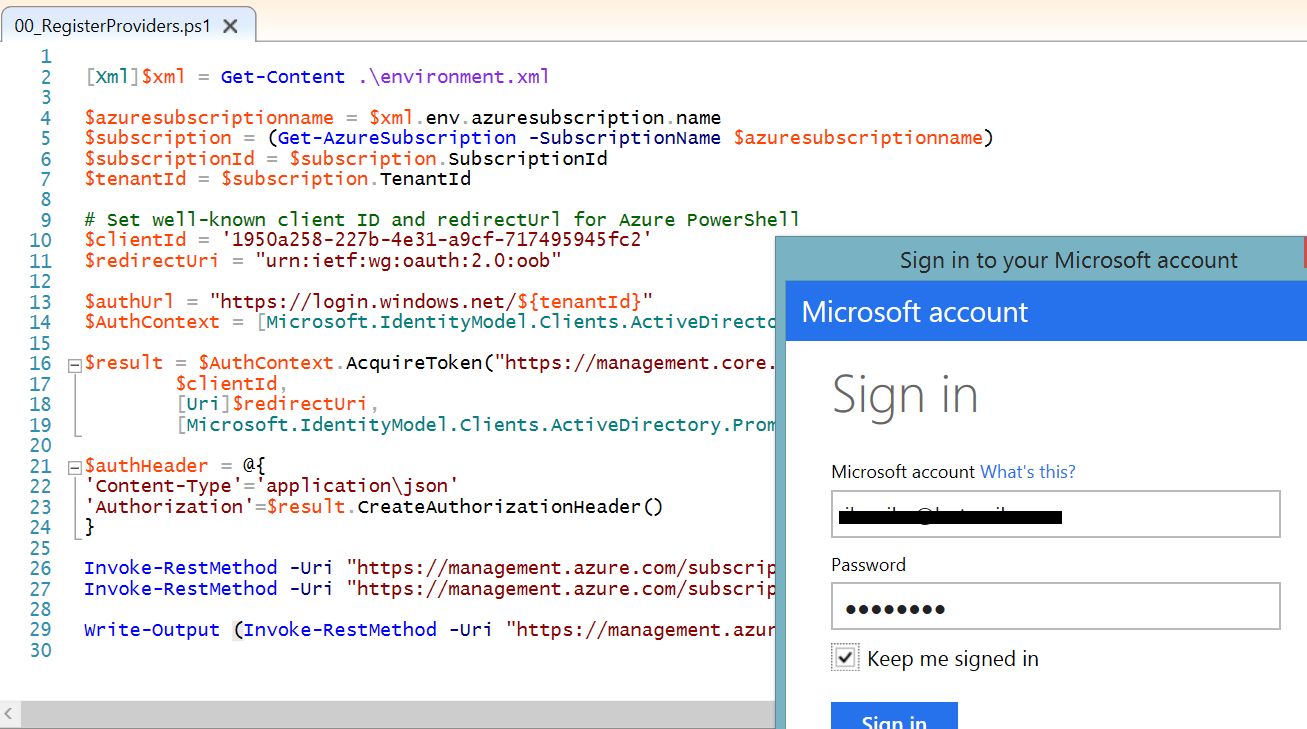
<documentdb name="myshuttletest" />

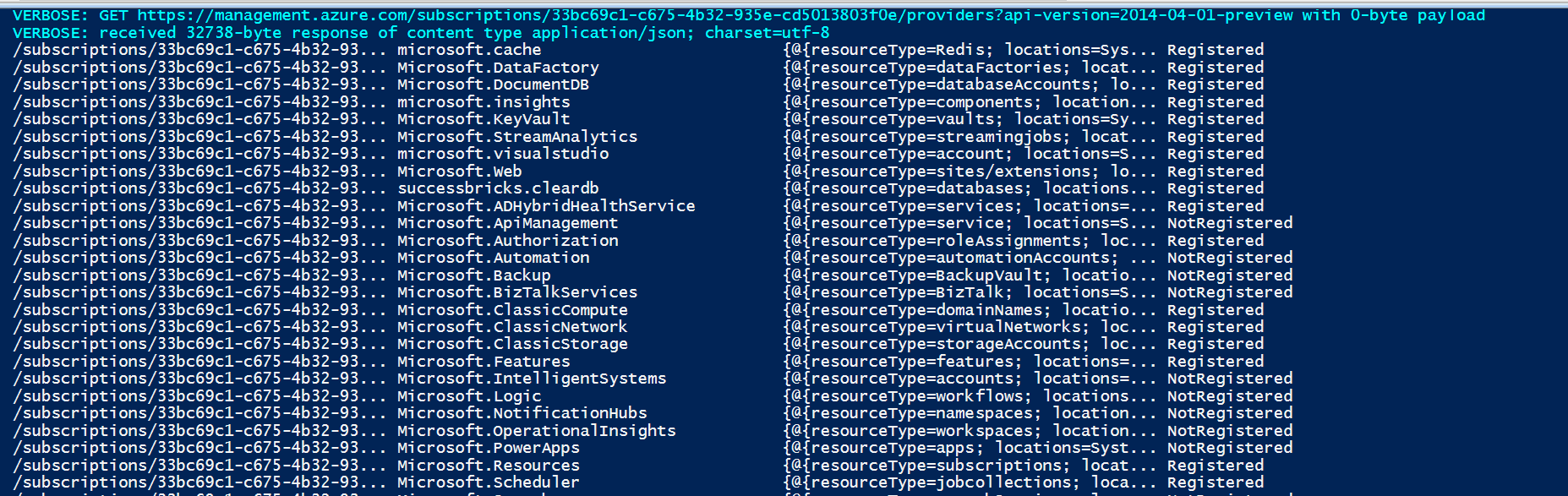
<websitewebapi name="myshuttletestWA" />

</env>

# Step 4. Register Azure Resources Providers

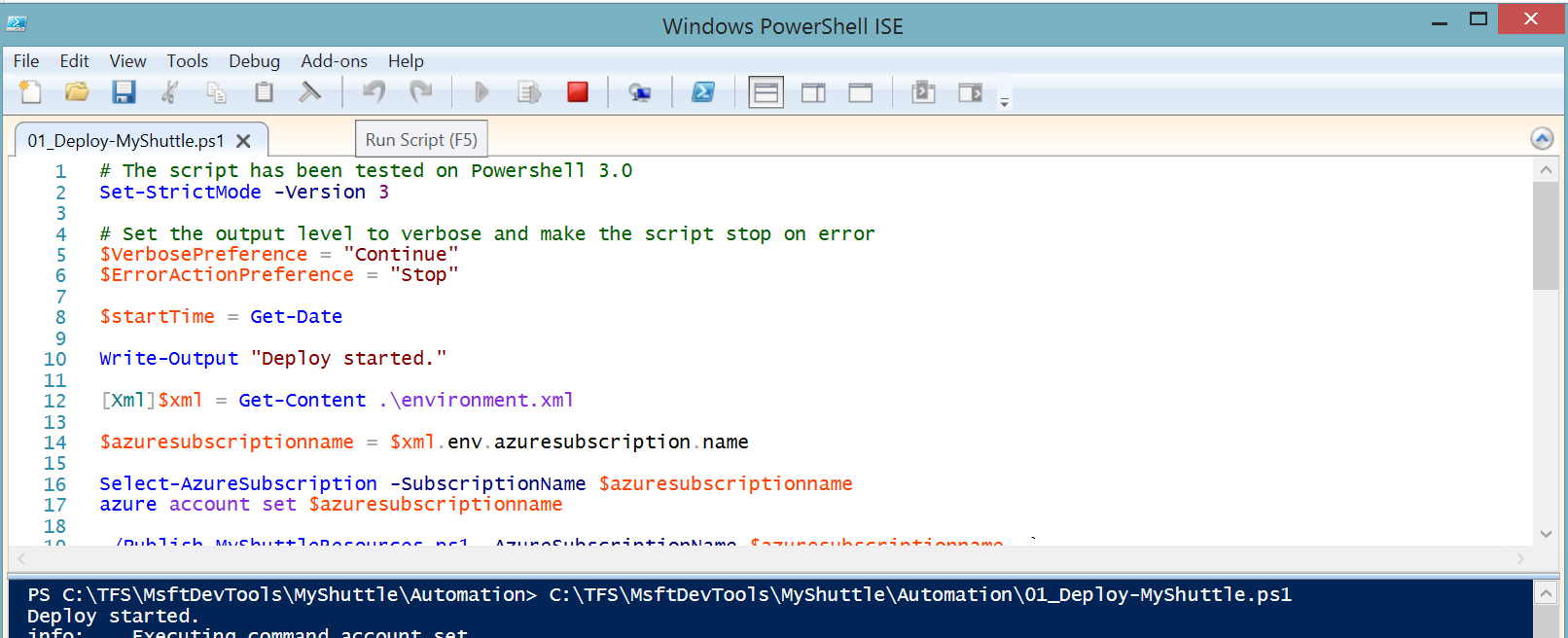
1. Open the file **00\_RegisterProviders.ps1** with Windows PowerShell ISE with admin privileges.
2. In Powershell ISE, position into the same folder where you have the “environment.xml” file with “cd folder\_path”
3. Run the script.



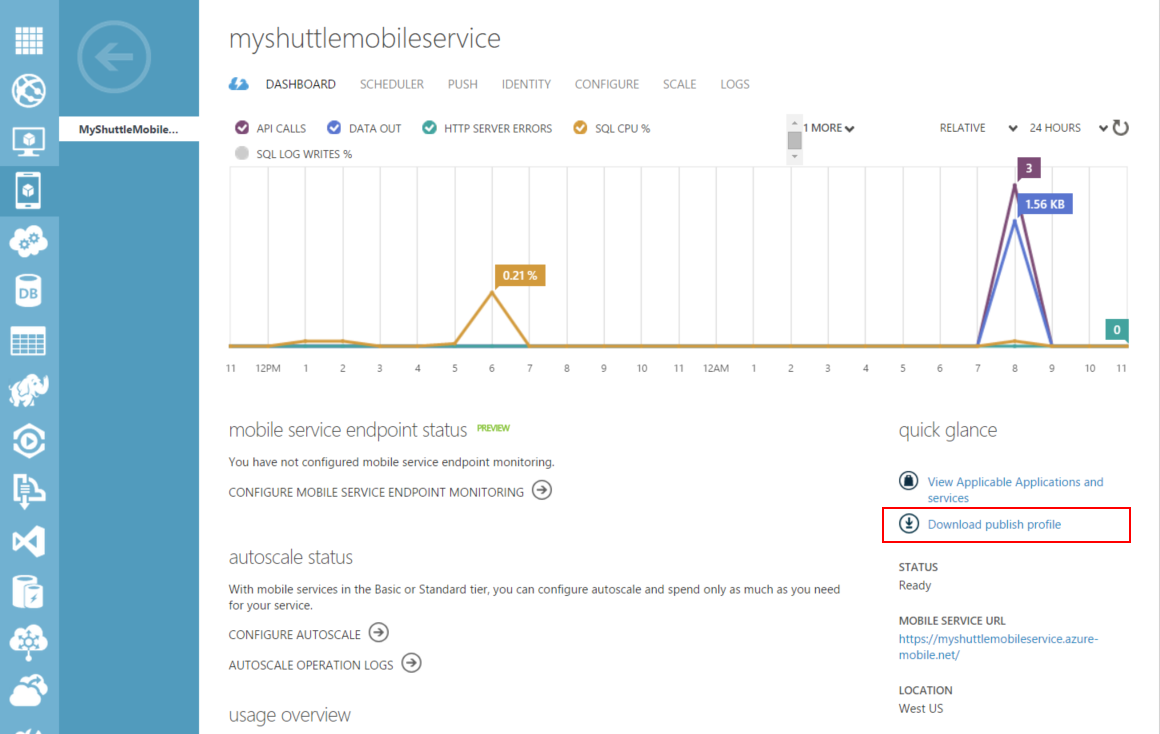


# Step 5. Create Azure resources for Apps demos

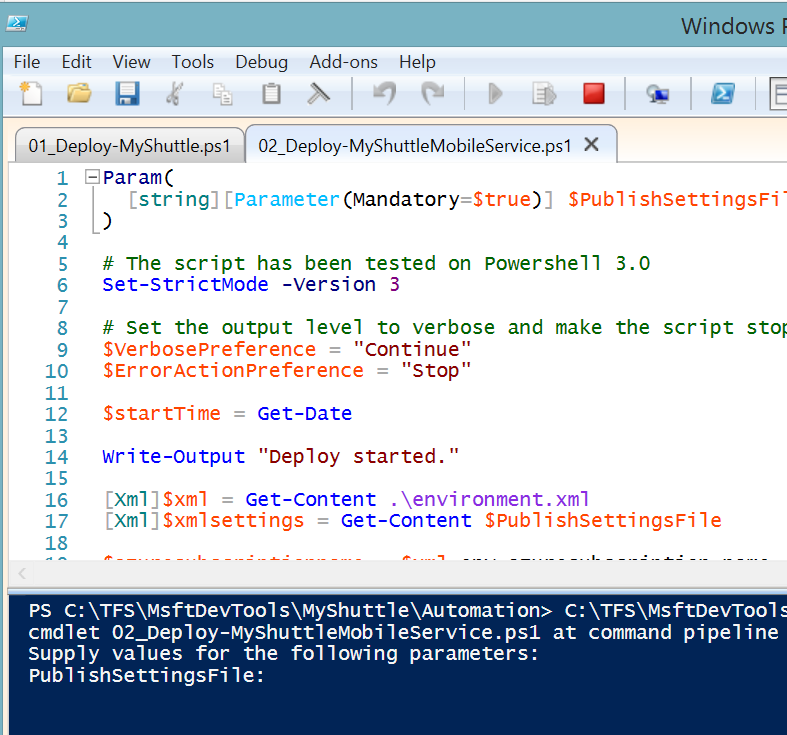
1. Open the file **01\_Deploy-MyShuttle.ps1** with Windows PowerShell ISE with admin privileges.
2. In Powershell ISE, position into the same folder where you have the “environment.xml” file with “cd folder\_path”
3. Run the script.



1. Login to the [Azure Management Portal](https://manage.windowsazure.com/). You can check the different assets created by the script, like Web Apps, SQL Database, etc.
2. In the Azure Management Portal, click on **Mobile Services** and select the mobile service that was defined in parameters file (websitemobileservice).
3. Go to Dashboard and click on Download publish profile.



1. Copy the file into the same folder where you have the .ps1 files for the automated deployments.
2. Open the file **02\_Deploy-MyShuttleMobileService.ps1** with Windows PowerShell ISE.
3. Run the script and set the name of the publish profile as parameter value.



Those two scripts would have created the following resources in your Azure subscription:

* **Azure SQL Database and SQL Database virtual Server**
  + YOUR\_SERVER.database.windows.net
  + USER
  + PASSWORD
* **Azure WebApp with the ASP.NET 5 application deployed**
  + <http://YOUR_SITE.azurewebsites.net/>web
  + AppSettings
    - Key => MyShuttleContext:ConnectionStringKey
    - Value =>
  + data source=**YOUR\_SERVER**.database.windows.net;initial catalog=MyShuttle;persist security info=True;user id=**USER**@**YOUR\_SERVER**;password=**PASSWORD**;MultipleActiveResultSets=True
* **Azure Storage Account named** **YOUR\_STORAGE**
* Public container named **myshuttleinvoice** with a PDF file (invoiceform.pdf)
* **WebApp with WebJob**
* Url: http://**YOUR\_SITE\_API**.azurewebsites.net/
* AppSettings:
  + Key => pdf::invoiceform
  + Value =><http://YOUR_STORAGE.blob.core.windows.net/myshuttleinvoice/invoiceform.pdf>
* Connection Strings:
  + Keys: AzureWebJobsDashboard & AzureWebJobsStorage.
  + Value: DefaultEndpointsProtocol=https;AccountName=**YOUR\_STORAGE**;AccountKey=ACCOUNT\_KEY
* **Mobile service**.
* <http://YOUR_SITE.azure-mobile.net/>
* AppSettings
  + Key: MS\_TableConnectionString
  + Data Source= **YOUR\_SERVER**.database.windows.net;Initial Catalog=MyShuttle;User ID= **USER**@**YOUR\_SERVER**;Password= **PASSWORD**;Asynchronous Processing=True;TrustServerCertificate=False;
  + Key: AzureWebJobsStorage.
  + Value: DefaultEndpointsProtocol=https;AccountName=YOUR\_STORAGE;AccountKey=ACCOUNT\_KEY

# Step 6. Create Azure resources for IoT and Big Data demos

1. Open 03\_Deploy-MyShuttle-IoT with Windows Powershell ISE.
2. In PowerShell ISE, move to the same folder where you have that Script.
3. Run the script.

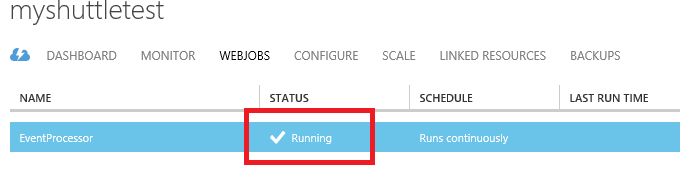
When creating the Service Bus resource, the script could raise an error message related to “EventHubExists” method. Close Windows Powershell ISE, open the script and run again it.

When creating the HDInsight cluster the script could raise an error message related to “**Unauthorized**” action. The script will be finish so, launch it again.

The ASP.NET portal Web App is created As Free. The websockets connection are limited to 5 so it could be a better option to change it to Shared.

* Free: (5) concurrent connections per website instance
* Shared: (35) concurrent connections per website instance
* **Basic: (350)** concurrent connections per website instance
* Standard: no limit

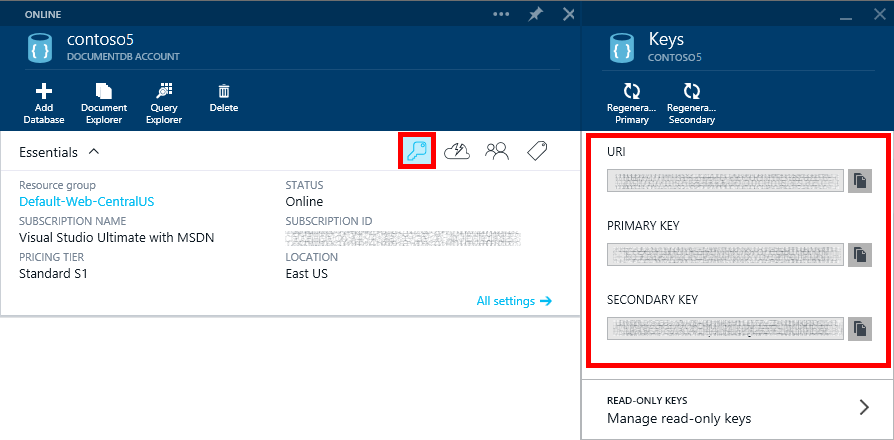
In this WebApp start or RESTART the webjob to be sure that it uses the right configuration.



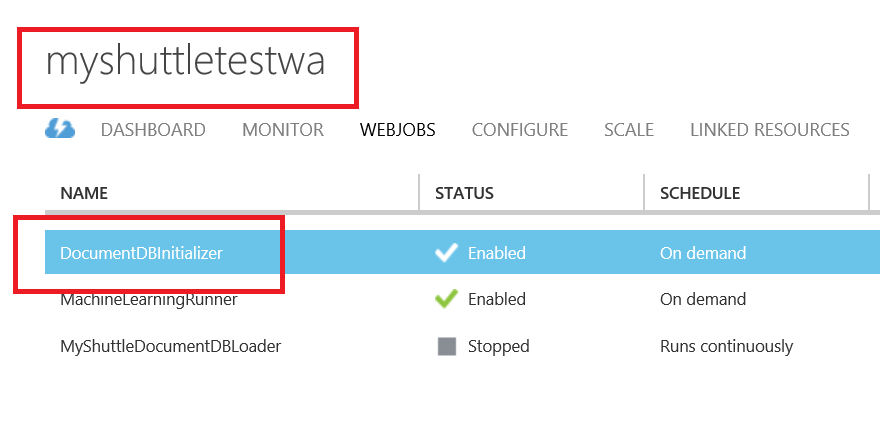
This script will create the following resources in your Azure subscription:

* Event Hub
  + Namespace
  + EventHub
* Stream Analytics
  + MyShuttle-Accelerometer
  + MyShuttle-Compass
  + MyShuttle-OBD
  + MyShuttle-OBD-SecurityBelt
  + MyShuttle-Rfid
* HDInsight
  + Cluster
* DataFactory
  + AccelerometerAggregatePipeline
  + AccelerometerAggregatePipeline\_output
* DocumentDB
  + <https://DOCUMENTDBNAME.documents.azure.com>
  + Database MyShuttle2
  + Collections: TrackedRides, VehiclesSummary, DrivingStyles
* WebAPI & WebJobs
  + <http://WEBSITEWEBAPINAME.azurewebsites.net/>web
    - "DocumentDb:EndpointUrl" => "https://DocumentDBNAME.documents.azure.com:443/";
    - "DocumentDb:AccessKey" = “[KEY”]
    - "DocumentDb:DatabaseId" = DOCUMENTDBNAME
    - "Settings:DeleteDocuments" => "true"
    - "HDinsight:EndpointUrl" = "https:// HDINSIGHTCLUSTERNAME.azurehdinsight.net";
    - "HDinsight:Login" = HDINSIGHTCLUSTERUSER
    - "HDinsight:Password" = HDINSIGHTCLUSTERPASSWORD

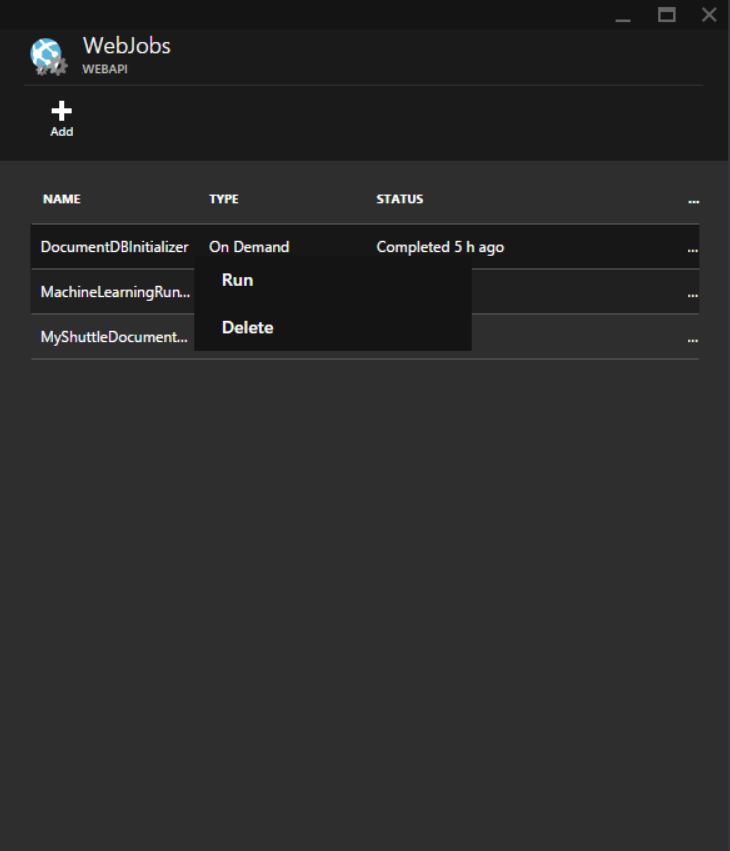
You need to set the valor of DocumentDb:AccessKey manually. This value is the PRIMARY KEY for your DocumentDB account, which can be obtained from the [Azure preview management portal](https://portal.azure.com/) blade for your DocumentDB account.



In order to initialize the DocumentDB database you need to run once the DocumentDBInitialize webjob.



1. In the Web App blade of the [Azure Portal](http://portal.azure.com/), click All settings > WebJobs to show the WebJobs blade.
2. To run the WebJob, right-click its name in the list and click Run.



# Step 7. Configure Apps

## Demo ASP.NET

The current ASP.NET app is working with the beta4 packages.

In Visual Studio you need to configure the nuget repositories to restore the packages. The beta4 packages were in <https://www.myget.org/F/aspnetrelease/api/v2> when this content was created.

## Demo Desktop App

* Solución: 01\_Demos\_NativeApps.sln
* Proyecto: MyShuttle.Client.Desktop
* App.config
  + URLPREFIXKEY => <http://YOUR_SITE.azurewebsites.net/>**web**

## Demo Native Apps

* Solución: 01\_Demos\_NativeApps.sln
* Windows Phone: MyShuttle.Client.UniversalApp.WindowsPhone
* Windows Store: MyShuttle.Client.UniversalApp.Windows
* Xamarin Android: MyShuttle.Client.Droid
* Xamarin iOS: MyShuttle.Client.iOS
* Configuración: (MyShuttle.Client.Core)
* Infrastructure>ApplicationSettingServiceSingleton.cs
  + DefaultUrlPrefixValue => <http://YOUR_SITE.azurewebsites.net/>web
  + DefaultBingMapsTokenValue => YOUR\_BING\_MAPS\_TOKEN\_VALUE

## Demo Connected Systems

* Solution: 05\_Demos\_Azure\_ConnServices.sln
* Open 01\_Demos\_NativeApps.sln to setup clients apps:
* Project - MyShuttle.Client.Core
  + Settings>CommonAppSettings.cs
    - \_MobileServiceUrl => http://YOUR\_SITE.azure-mobile.net/
    - \_MobileServiceKey => YOUR\_MOBILE\_SERVICE\_KEY
    - \_SignalRUrl => http://YOUR\_SITE.azurewebsites.net
  + Infrastructure>ApplicationSettingServiceSingleton.cs
    - DefaultUrlPrefixValue => <http://YOUR_SITE.azurewebsites.net/web>
    - DefaultBingMapsTokenValue => YOUR\_BING\_MAPS\_TOKEN\_VALUE

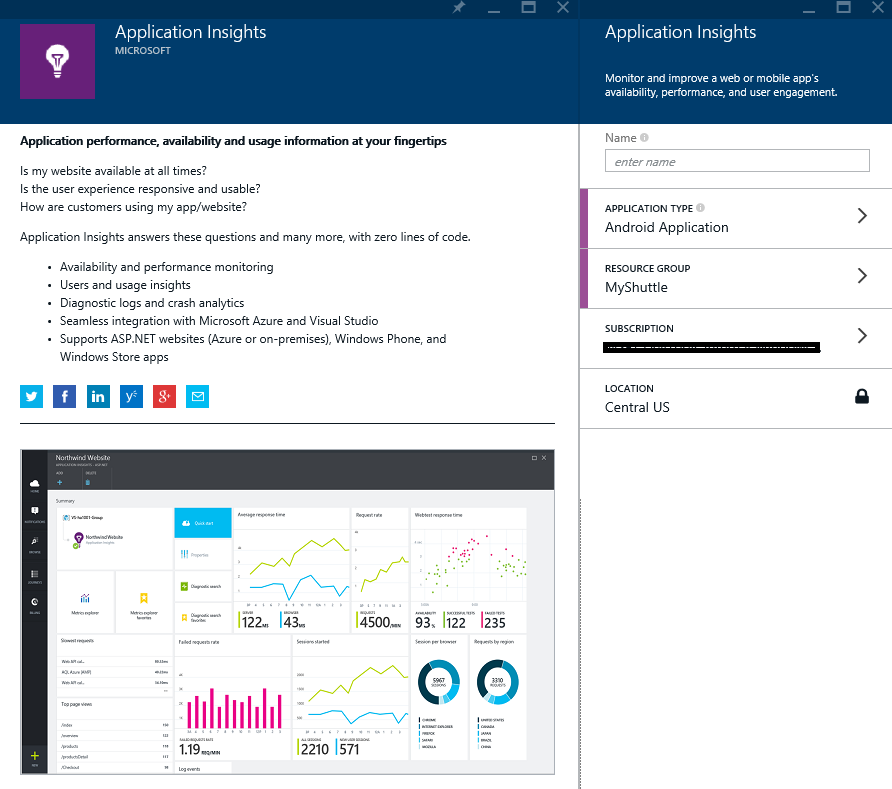
## Demo Cordova

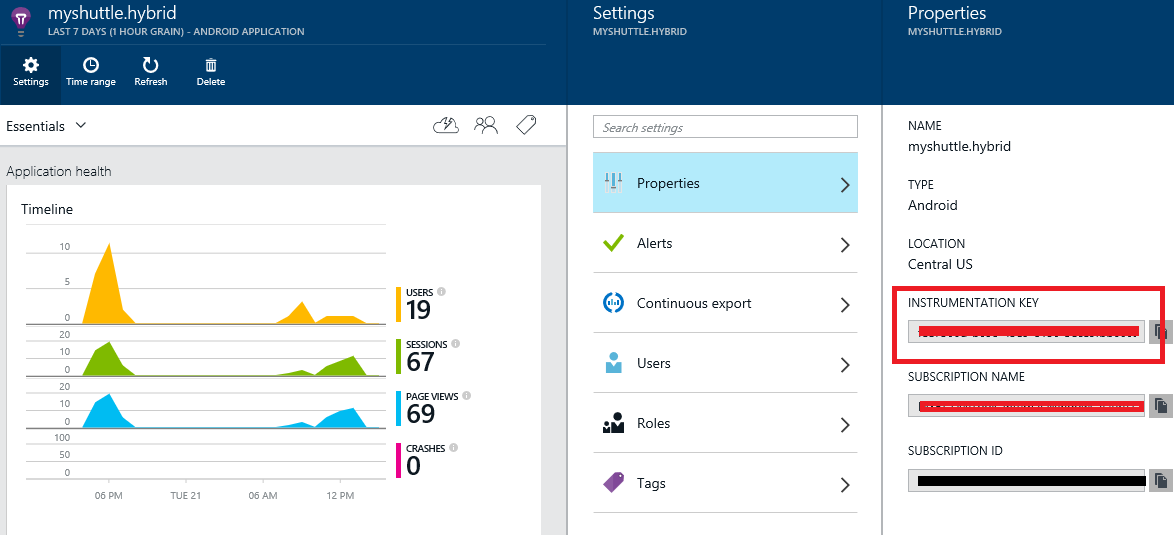
* Solution: 03\_Demos\_Cordova.sln
* Before running the app, you need to run this command lines:
  + npm install bower –g
  + npm install gulp –g
  + npm install karma-cli –g
* In the app root directory (src\MyShuttle.Client.Cordova.New)
  + npm install
  + gulp beforeBuild
* Set in scripts>modules\core\services>settingsService.ts
  + this.bingMapsKey = 'YOUR\_BING\_MAPS\_TOKEN\_KEY';
  + this.realTimeNotificationsServerUrl = 'http://YOUR\_SITE.azurewebsites.net/**web'**;
  + this.mobileServiceKey = 'YOUR\_MOBILE\_SERVICE\_KEY';

this.storageService.getValue('serviceUrl', 'http://YOUR\_SITE.azure-mobile.net/');

* AppInsights.
  + In your Windows Azure subscription, create a new Application Insights Service.
  + Edit the file scripts\index.ts and set the instrumentation key:

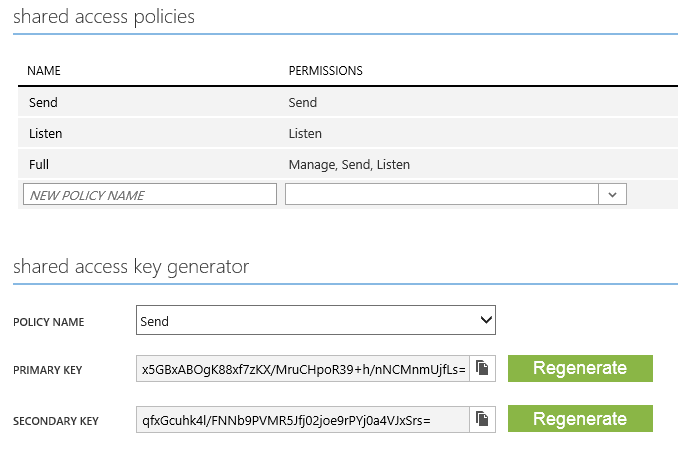
appInsights.config.instrumentationKey = "YOUR\_KEY";





## Demo IoT

* Solution MyShuttle\_IoT.sln
  + Device; MyShuttle.Device project
    - Configure EventHub connectionString. Use the Send or Full access policy created in the EventHub.
    - Add key="Microsoft.ServiceBus.ConnectionString" value="Endpoint=sb://xxxxx-ns.servicebus.windows.net/;SharedAccessKeyName=**Send**;SharedAccessKey=[key]" />



* + - Configure EventHubName

<setting name="EventHubName" serializeAs="String">

<value>EVENT\_HUBNAME</value>

</setting>

* + Windows Store; MyShuttle.Dashboard.Client>Constants.cs
    - Azure Api URI.

## Demo Dashboard

* Solution MyShuttle\_IoT.sln
  + MyShuttle.Dashboard.Client project
    - Open Constants.cs file and set the URL of the WebAPI service.

public static class Constants

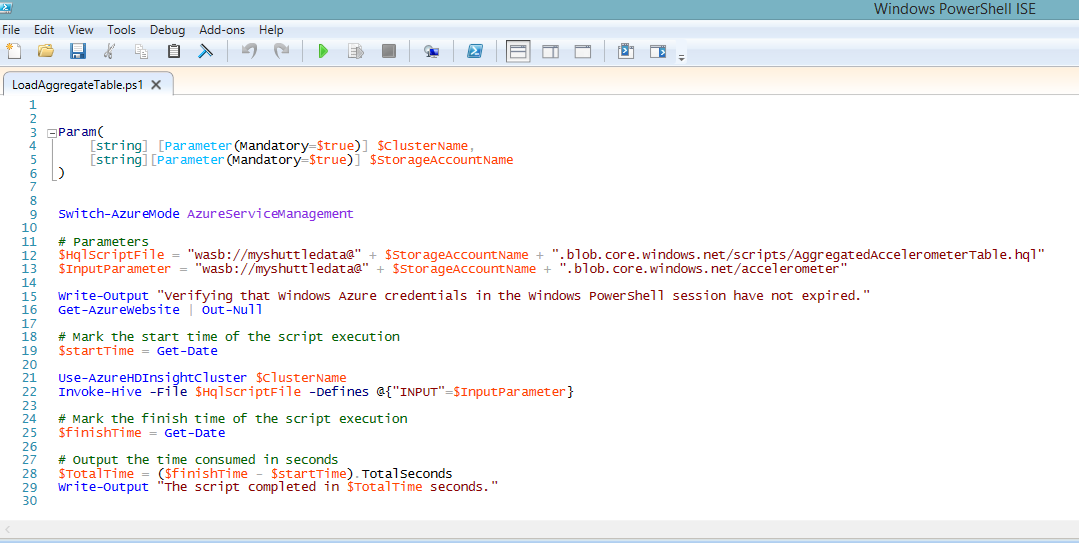
{

public const string ServerAddress = "http://WEBAPI.azurewebsites.net";

}

# Step 8. Initialize the accelerometer aggregate Hive table

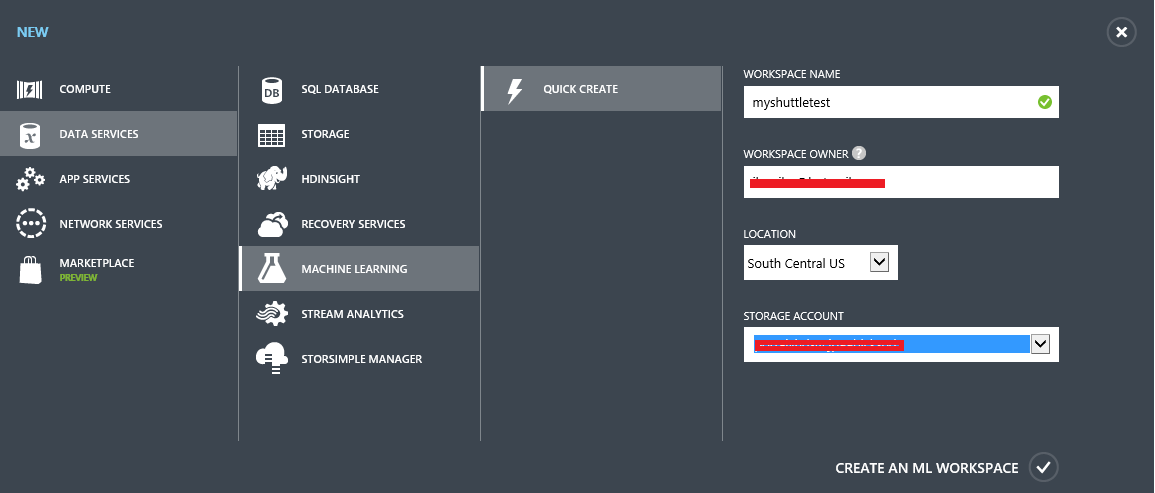
1. Open the file **LoadAggregateTable.ps1** with Windows PowerShell ISE with admin privileges. The script is in the “Automation\IoTBigData\DataFactory” directory.
2. Run the script.



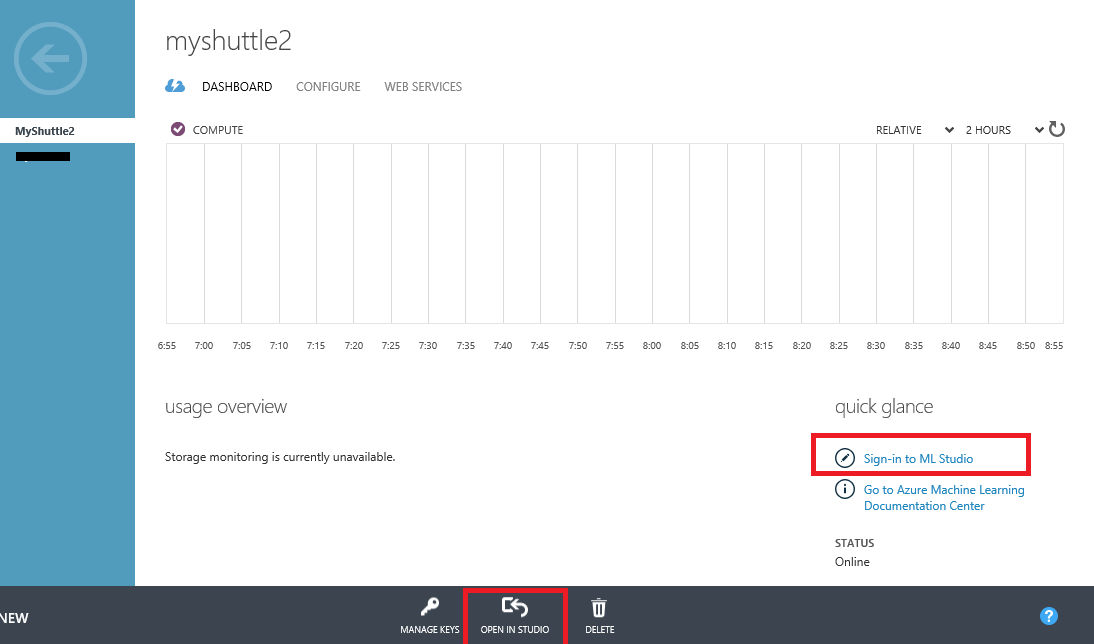
1. The script will be ask for the cluster and storage name configured in your environment.xml

# Step 9. Create the Machine Learning experiments

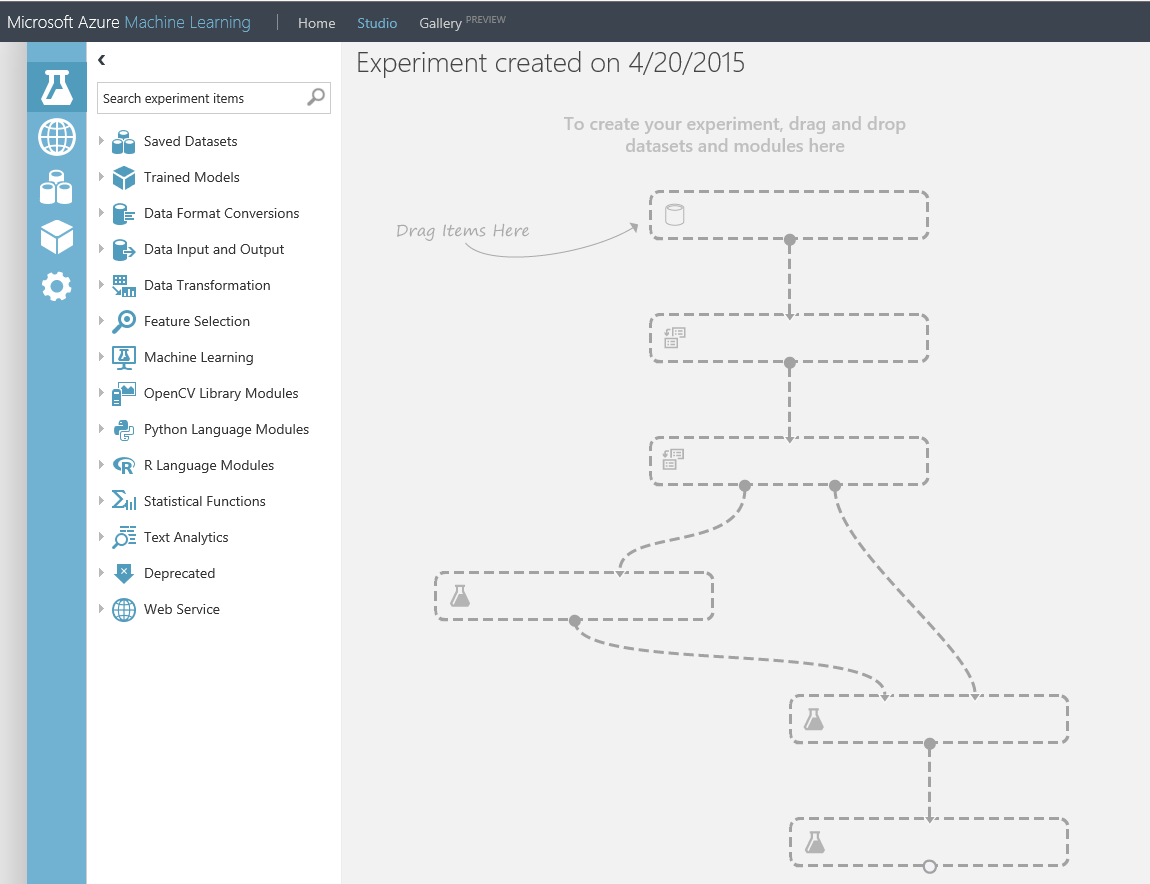
1. Go to the Azure Portal and create a new Machine Learning service.



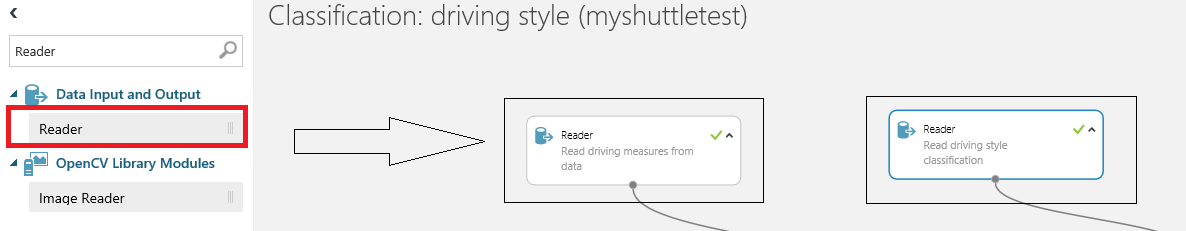
1. After creating the service, go the service dashboard.
2. Open the ML Studio.



1. Start a new experiment by clicking +NEW at the bottom of the Machine Learning Studio window, select EXPERIMENT, and then select "Blank Experiment".
2. Select the default experiment name at the top of the canvas and rename it to something meaningful, for example, **Classification: driving style**



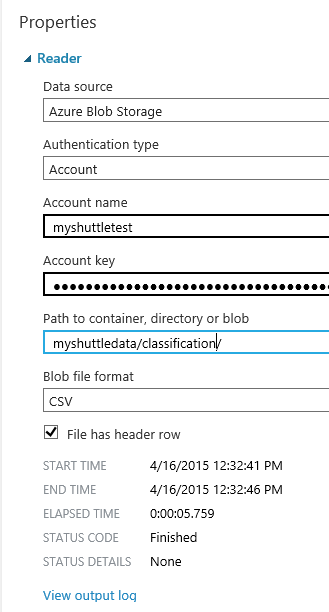
1. To the left of the experiment canvas is a palette of datasets and modules.
2. Type “Reader” in the search box at the top of this palette.
3. Drag two readers to the experiment canvas.



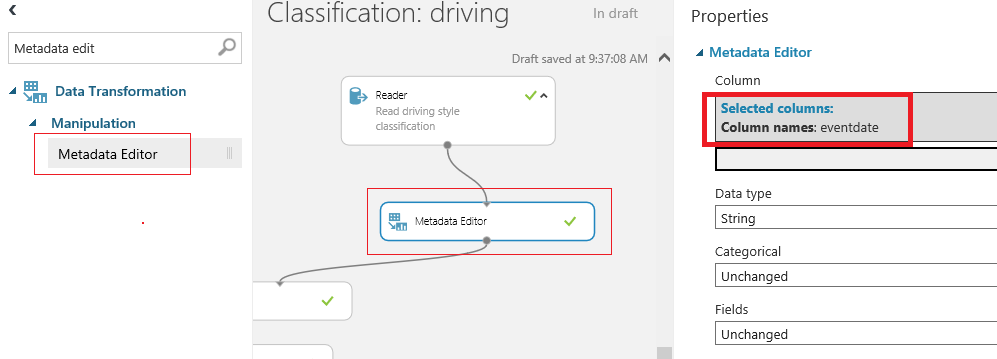
1. Click on the left reader and configure it in the right panel.
   1. Data Source: Hive Query
   2. Hive Database Query:
      1. select \* from accelerometeraggregateoutput
   3. HCatalog server URI
      1. <https://your_server.azurehdinsight.net>
      2. The name is configured in the environment.xml
   4. Hadoop user account name
      1. The user name configured in the environment.xml.
   5. Location outputdata
      1. The password configured in the environment.xml.
   6. Azure storage account name
      1. The name is configured in the environment.xml
   7. Azure storage key
      1. Get it from the azure portal.
   8. Azure container name
      1. Your cluster name, for example, myshuttletest.
      2. The name is configured in the environment.xml



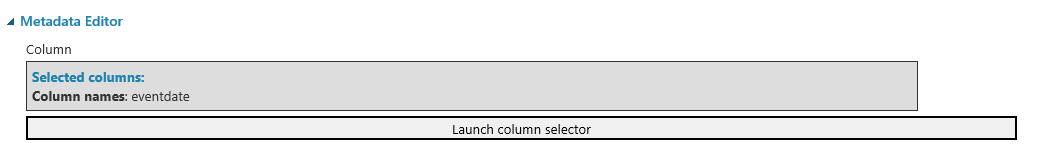
1. Click on the right reader and configure it in the right panel.
   1. DataSource.
      1. Azure Blob Storage
   2. Authentication type
      1. Account
   3. Account name.
      1. The name is configured in the environment.xml
   4. Account key.
      1. Get it from the azure portal
   5. Path to container.
      1. myshuttledata/classification/
   6. Blob file format
      1. CSV
   7. File has header row
      1. Checked

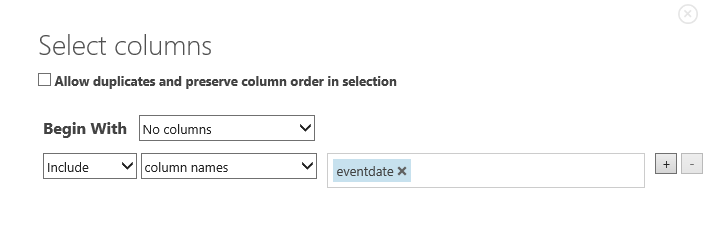


1. Type “MetadaEditor” in the search box at the top of this palette.
2. Drag it to the experiment canvas.
3. Connect the Blob reader to the Metadata Editor

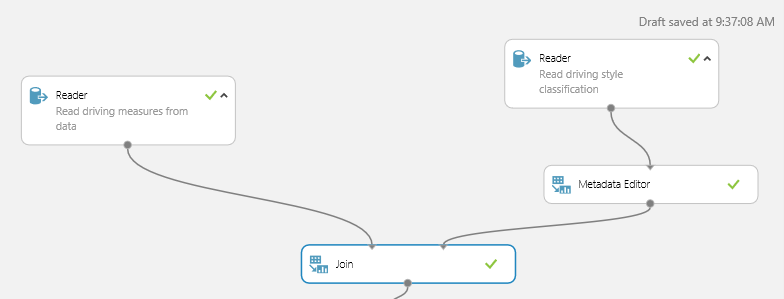


1. Click on “Launch column selector” and select **eventdate**.

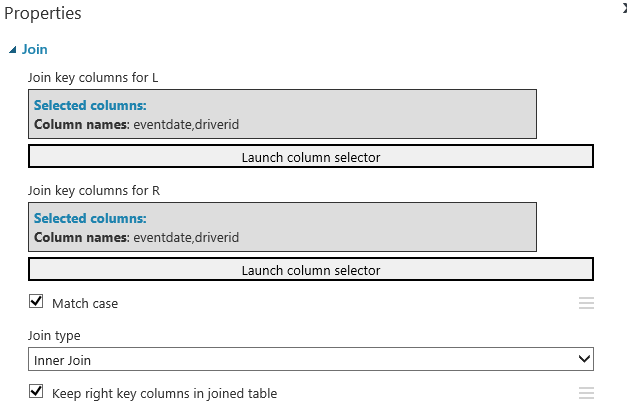




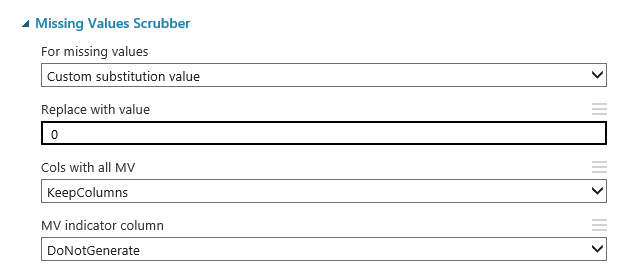
1. Type “Join” in the search box at the top of this palette.
2. Drag it to the experiment canvas.
3. Connect the Join box to the two existing readers.



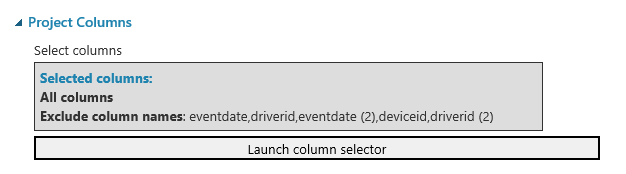
1. Configure the join box.



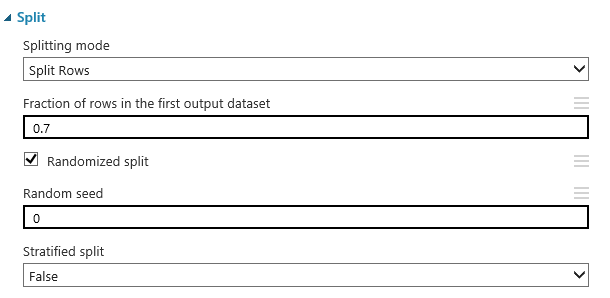
1. Drag and connect:
   1. Missing Values Scrubber

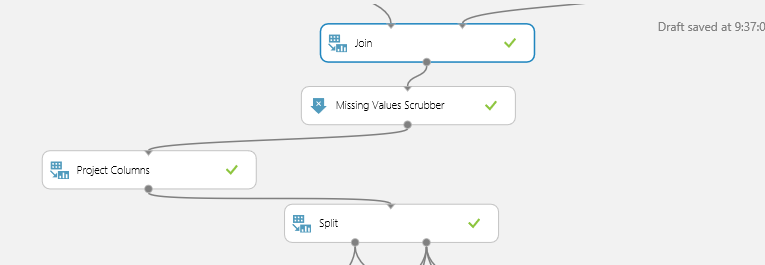


* 1. Project Columns.

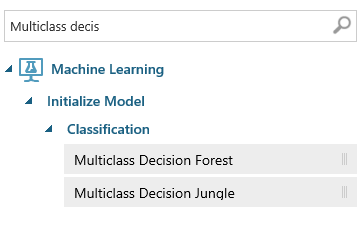


* 1. Split

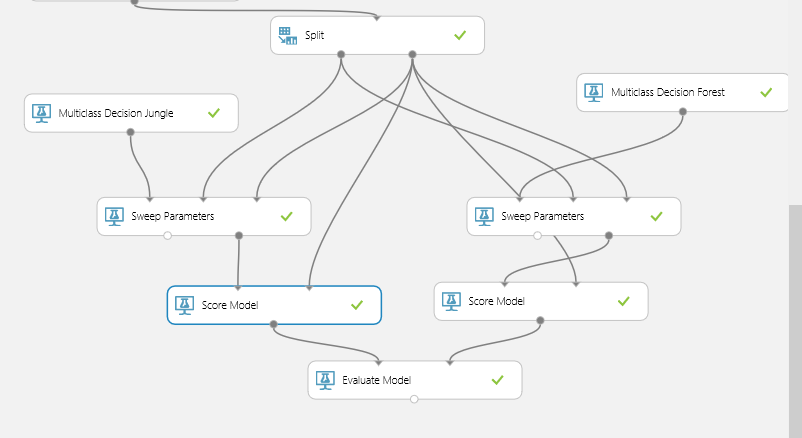


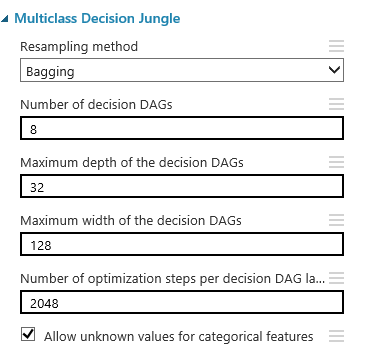


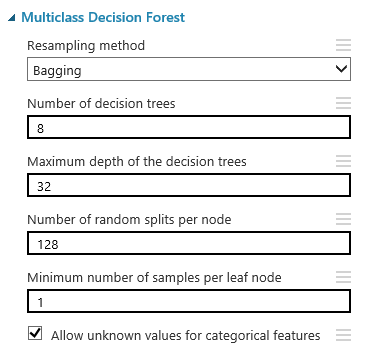
1. Type “Multiclass decision” in the search box at the top of this palette.
2. Drag them to the experiment canvas



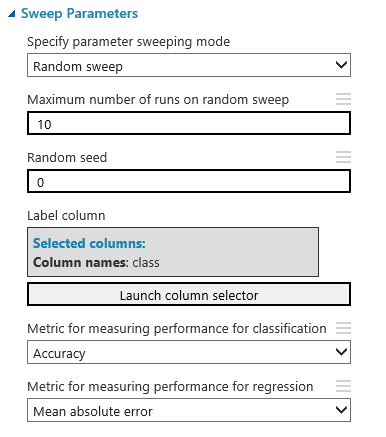
1. The result will be:



1. Set multiclass decision jungle properties
2. Set multiclass decision forest properties



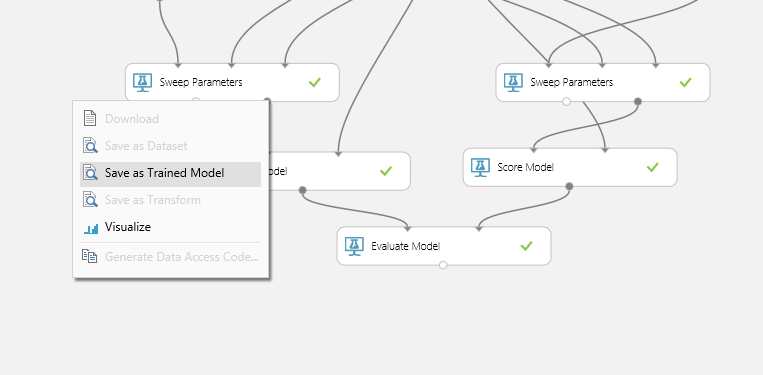
1. Set “Sweep Parameters” properties. In the two boxes set the same values.



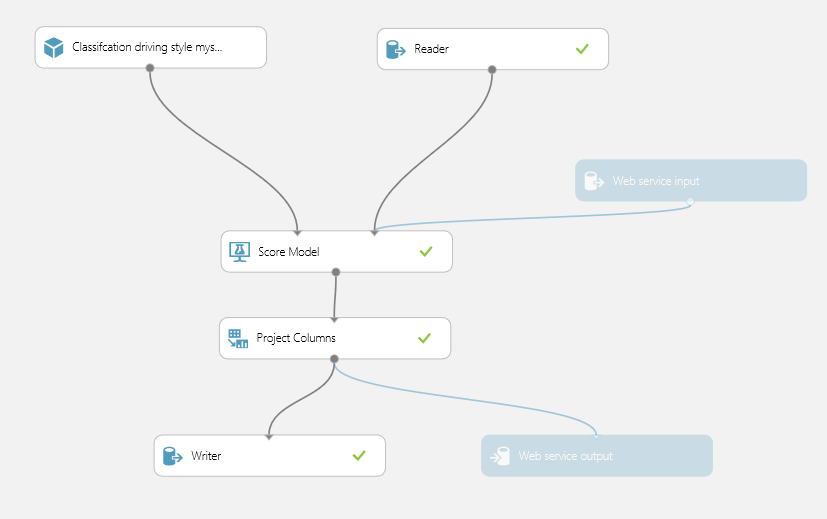
1. Score Model and Evaluate model don´t need properties.
2. Save the experiment.
3. Run it.



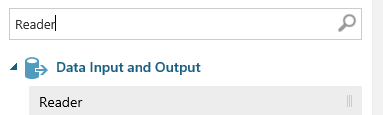
1. Save the experiment as Trained Model. Name “Classifcation driving style”



1. Start a new experiment by clicking +NEW at the bottom of the Machine Learning Studio window, select EXPERIMENT, and then select "Blank Experiment".
2. Select the default experiment name at the top of the canvas and rename it to something meaningful, for example, **Driver style classificatory**



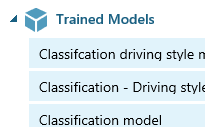
1. Type “Reader” in the search box at the top of the palette.



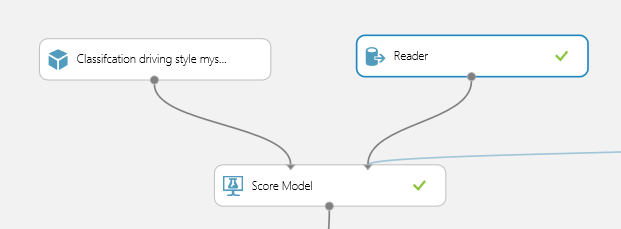
1. Drag it to the experiment canvas.
2. Click on the left reader and configure it in the right panel.
   1. Data Source: Hive Query
   2. Hive Database Query:
      1. select \* from accelerometeraggregateoutput
   3. HCatalog server URI
      1. <https://your_server.azurehdinsight.net>
      2. The name is configured in the environment.xml
   4. Hadoop user account name
      1. The user name configured in the environment.xml.
   5. Location outputdata
      1. The password configured in the environment.xml.
   6. Azure storage account name
      1. The name is configured in the environment.xml
   7. Azure storage key
      1. Get it from the azure portal.
   8. Azure container name
      1. Your cluster name, for example, myshuttletest.
      2. The name is configured in the environment.xml



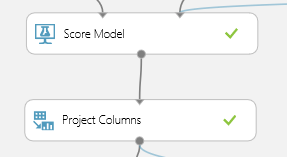
1. Type “Classifcation driving style” in the search box at the top of the palette



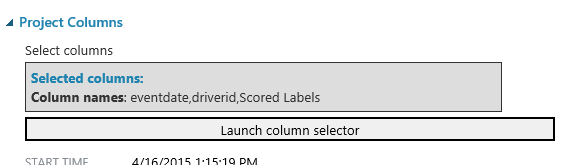
1. Drag your trained model to the experiment canvas.
2. Click on the left reader and configure it in the right panel.
3. Drag a new Score Model and connect to the existing boxes.



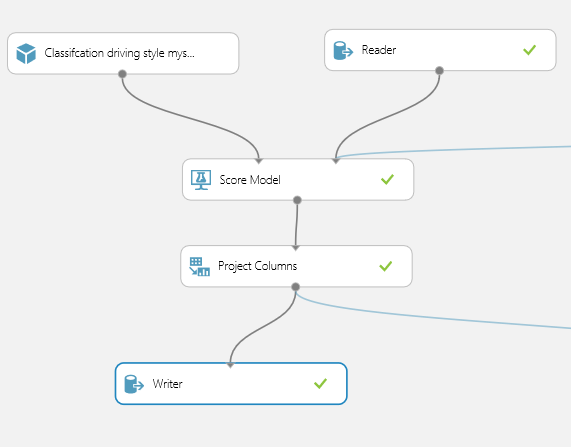
1. Drag a “Project Columns” box and connect to the ScoreModel box.



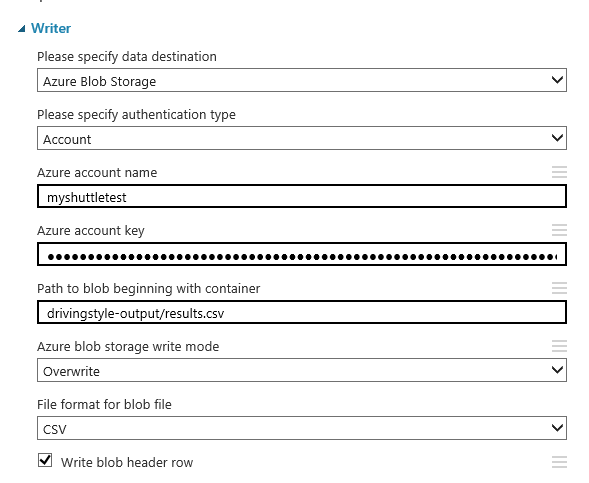
1. Set Project Columns properties.



1. Drag a “Writer” box



1. Set Writer properties
   1. The Azure account name is the storage name configured in your environment.xml



1. Save the experiment.
2. Run it.
3. Publish Web Service.

