## 環境準備

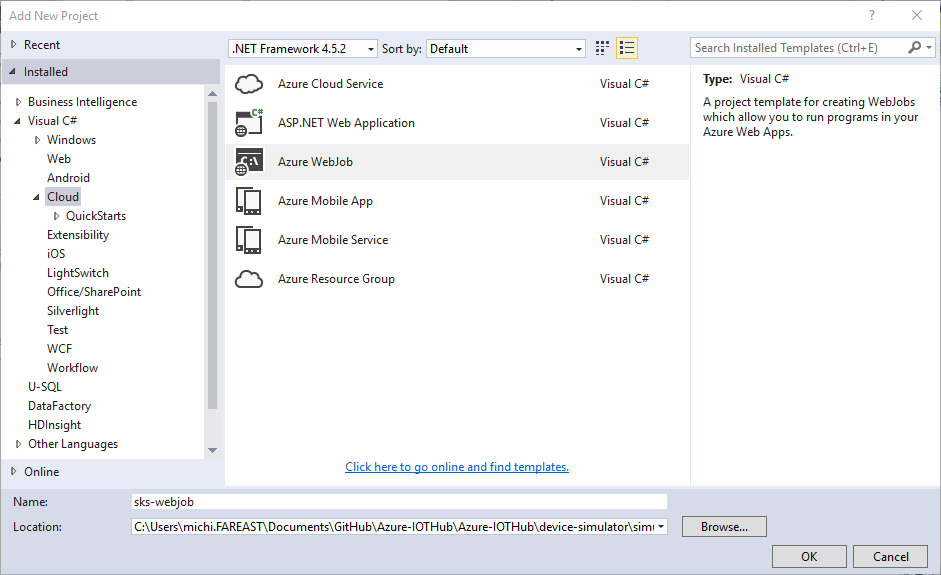
* Visual Studio 2015
* Azure SDK

## 注意

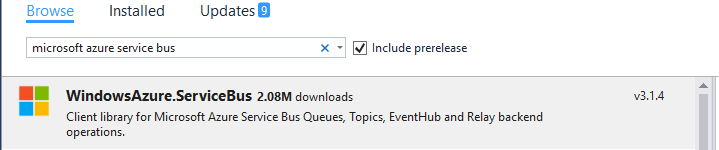
* 此方式僅適用於Stream Analytics輸出的Queue Message

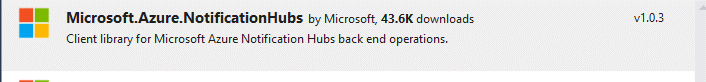
## 步驟

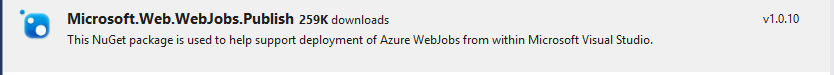
* 已Administrator權限打開Visual Studio 2015，新增一個 Web Job專案



* 以NUGet新增以下參考





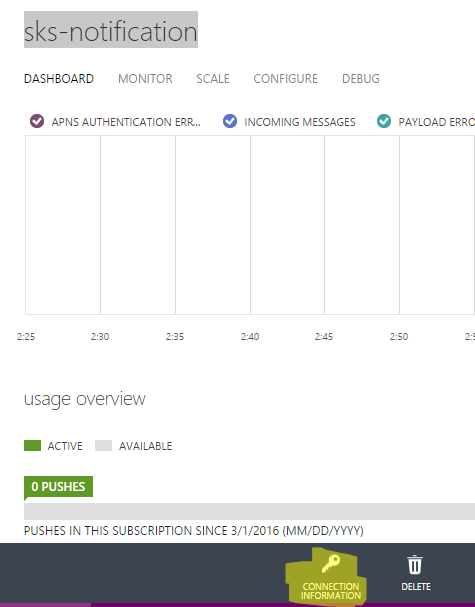


* 打開Functions.cs檔案，加入以下的namespace

using Microsoft.Azure.NotificationHubs;

using Microsoft.ServiceBus.Messaging;

* 新增以下的Function；其中<hub name>為先前建立的notification hub名稱；Connection String為Full Access的Connection String，可以在以下找到



* 在Functions.cs新增namespace引用

using Microsoft.Azure.WebJobs;

using Microsoft.Azure.NotificationHubs;

using Newtonsoft.Json;

* 在Functions.cs新增以下程式碼

private static string nhConnectionString = "Endpoint=sb://skshub.servicebus.windows.net/;SharedAccessKeyName=DefaultFullSharedAccessSignature;SharedAccessKey=+toF6XlnEcBSPlDTN0WPW2XBNmpIEQnZbQUTfO2y/aY=";

private static string hubName = "sks-notification";

private static async Task SendNotificationAsync(string msg)

{

await SendWindowsNotificationAsync(msg);

await SendAndroidNotificationAsync(msg);

}

private static async Task SendWindowsNotificationAsync(string msg)

{

NotificationHubClient hub = NotificationHubClient

.CreateClientFromConnectionString(nhConnectionString, hubName);

var toast = $"<toast launch=\"launch\_arguments\"><visual><binding template=\"ToastText01\"><text id=\"1\">{msg}</text></binding></visual></toast>";

var results = await hub.SendWindowsNativeNotificationAsync(toast);

}

private static async Task SendAndroidNotificationAsync(string msg)

{

NotificationHubClient hub = NotificationHubClient

.CreateClientFromConnectionString(nhConnectionString, hubName);

Newtonsoft.Json.Linq.JObject o = JsonConvert.DeserializeObject(msg) as Newtonsoft.Json.Linq.JObject;

var toast = "{data:{message:'{device} alert at {time}'}}".Replace("{device}", (string)o["deviceid"]).Replace("{time}", (string)o["time"]);

var results = await hub.SendGcmNativeNotificationAsync(toast);

}新在Functions.cs增以下程式碼，其中inputqueue為先前在Stream Analytics中定義輸出的Queue name

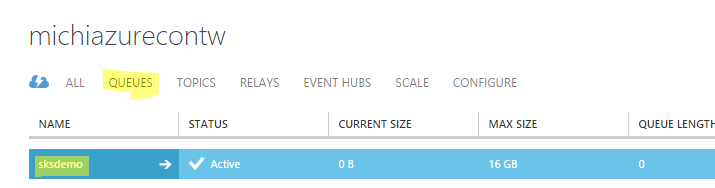
public static void ProcessQueueMessage([ServiceBusTrigger("sksdemo")] string message,

TextWriter logger)

{

logger.WriteLine(message);

SendNotificationAsync(message);

} 

* 修改Main()如下：

static void Main()

{

JobHostConfiguration config = new JobHostConfiguration();

config.UseServiceBus();

JobHost host = new JobHost(config);

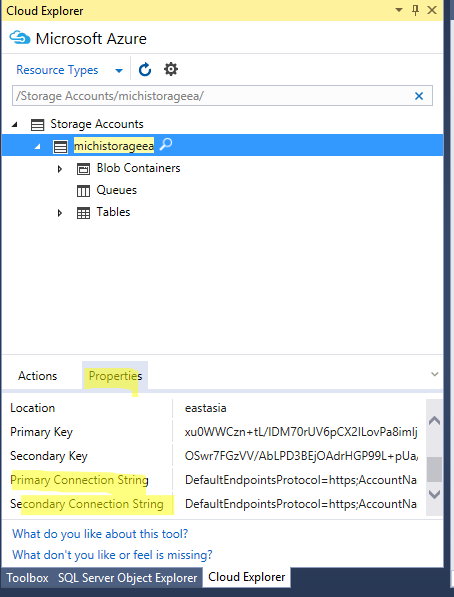
host.RunAndBlock();

}

* 打開App.Config，修改以下兩個Connection String；此兩個Storage為提供Web Job Log以及程式使用；你可以在Visual Studio的Cloud Explorer中取得Connection String

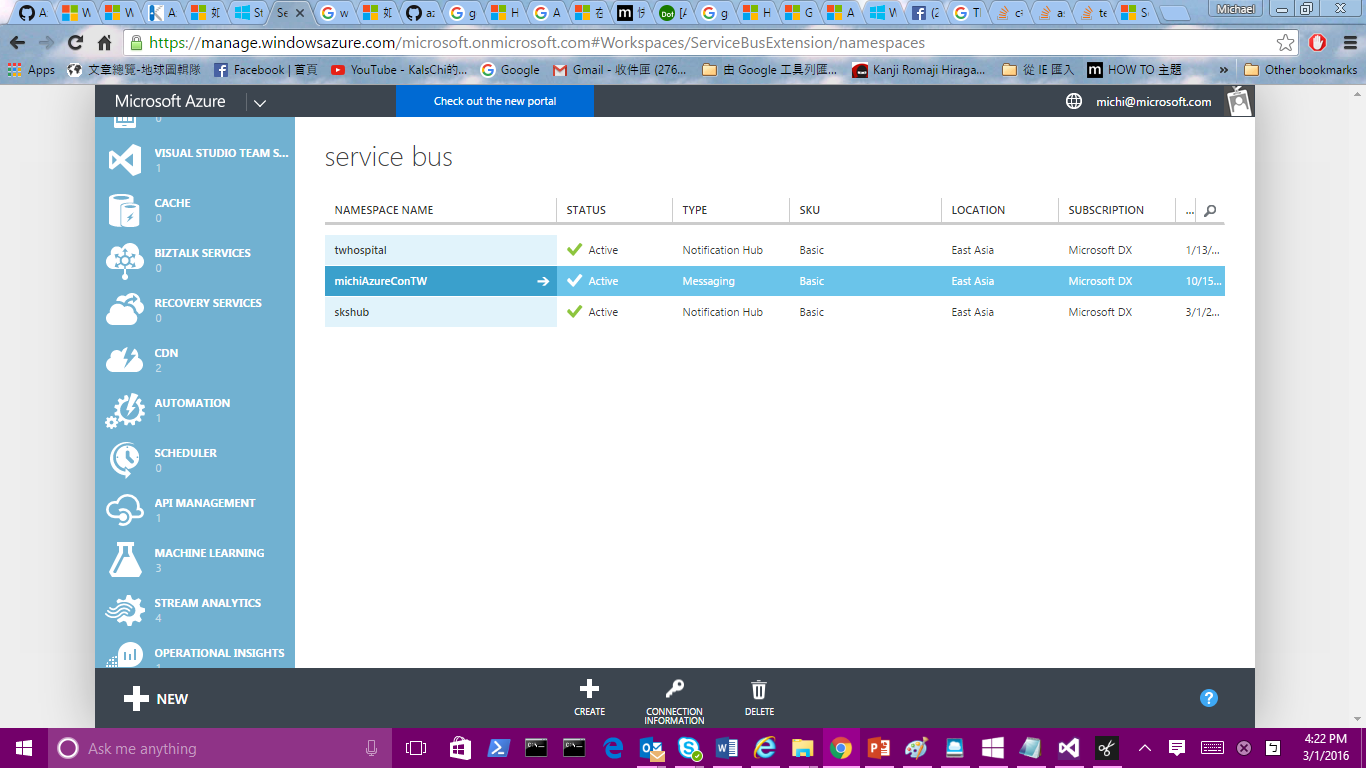
<add name="AzureWebJobsDashboard" connectionString="" />

<add name="AzureWebJobsStorage" connectionString="" />

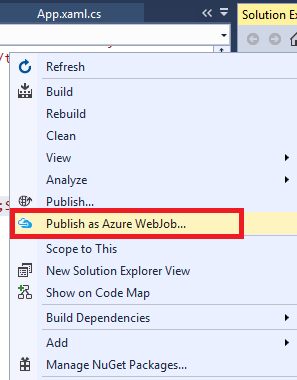


* 新增以下的Service Bus Connection String；此連線字串可在Service Bus層級的connection info取得

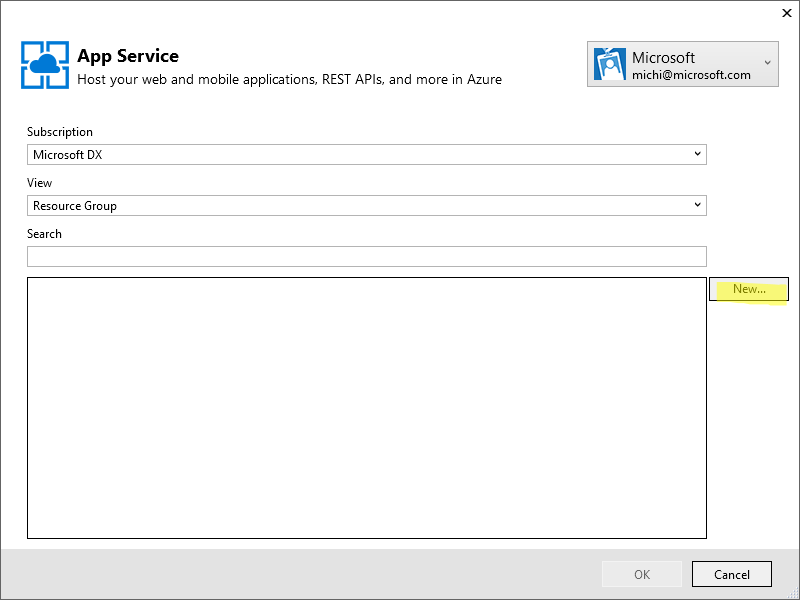
<add name="AzureWebJobsServiceBus" connectionString="Endpoint=sb://[yourServiceNamespace].servicebus.windows.net/;SharedAccessKeyName=RootManageSharedAccessKey;SharedAccessKey=[yourKey]"/>



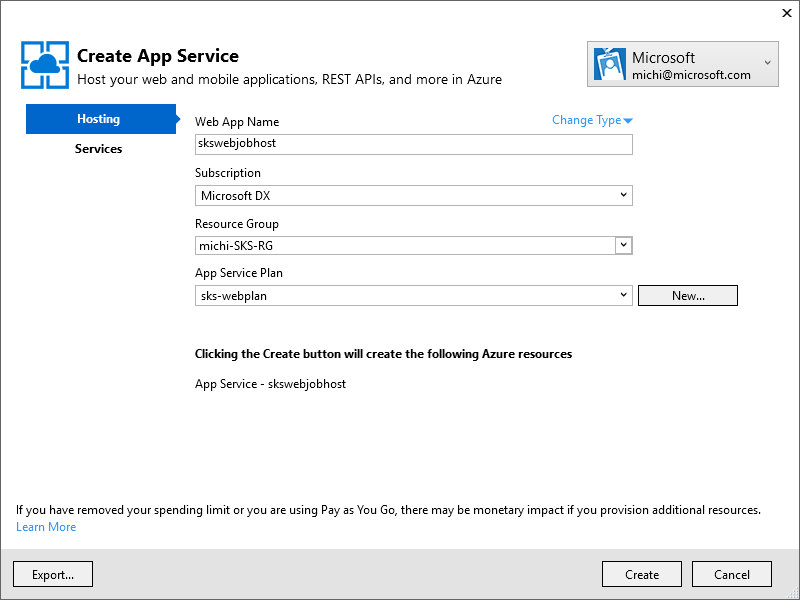
* 發布此Web Job



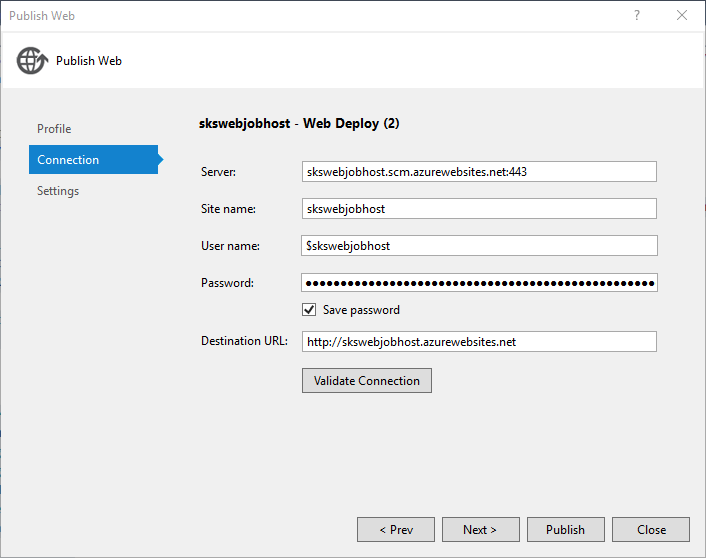
* 選擇新增一個Web Job



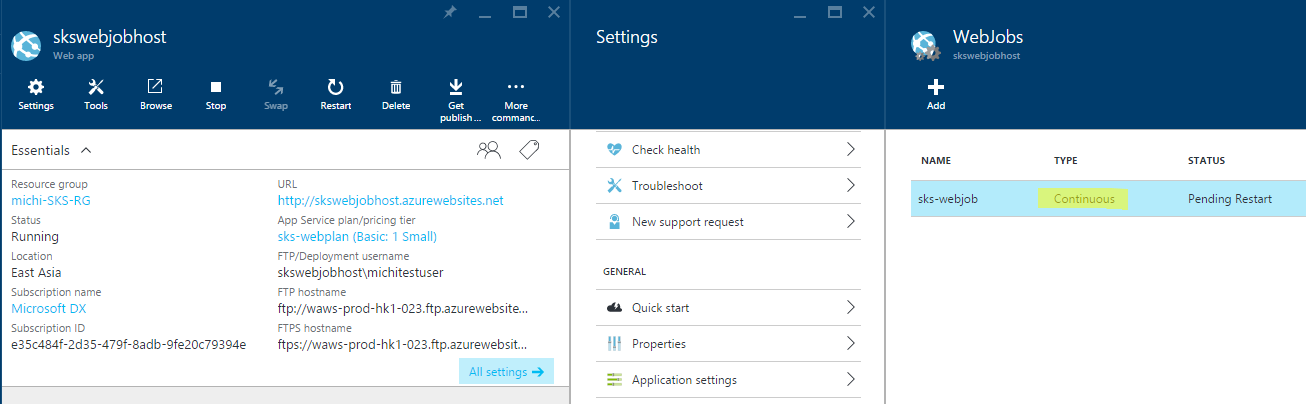
* 輸入基本資料



* 按下Publish發布



* 回到Portal可以看到剛剛的Web Job已經建立



## 參考資料

* <https://azure.microsoft.com/zh-tw/documentation/articles/websites-dotnet-webjobs-sdk-service-bus/>