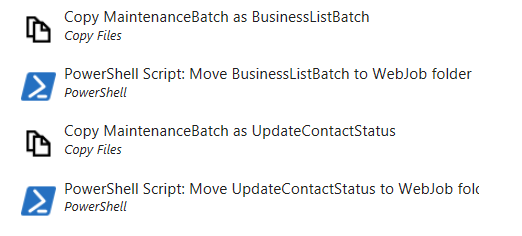
This post was published to Code Blog at 2:22:00 PM 6/20/2017

Creating Multiple Azure WebJobs from the Same VS Console Project

Category .NET ; Azure ; C# ; VSTS

​During the CJM project we had to create multiple Azure WebJobs.  Majority of them were fairly large and made sense to have their own unique VS project.  But, towards the end of the project there was the need for smaller, maybe even one-off Azure WebJobs.  So instead of the overhead of creating multiple VS console projects for each individual Azure WebJob, I worked with Eric Hutchinson to be able to create mutiple Azure WebJobs from a single VS console project.

Currently each time a new Azure WebJob is needed, steps will have to be manually added to the release.  In the future, Eric is going look into creating a custom scrip that will look for all .cmd files in the directory and repeate the process for each Azure WebJob.



# How to Set up

See $/Market Sell/CustomerJourney/Batch/Main/CustomerJourneyBatch/MaintenanceBatch for an example

There are 3 files in the VS console project that are needed to accomplish this.

* Program.cs (Normal launch point for console projects)
* .cmd file for each Azure WebJob
* .job file for each Azure WebJob

## Program.cs

Logic

* Include the normal string[] args in the parameters of the Main method
* Add a switch or if/else block to handle which job to execute based on the parameters passed in

## .CMD File

Naming: <Azure WebJob Name>.cmd

Logic:

* <VS console project name>.exe "<Azure WebJob Name>"

This will launch the console project passing the webjob name into the main method of the program.cs

\*Make sure this file is created outside of Visual Studio or else Visual Studio will include extra BOM character that won't allow Azure to run the job.

\*Set the build action on the files to be "Content"

## .Job File

Naming:  <Azure WebJob Name>.job

Logic:

* Sets up the scheduling for the job if needed.
* Enter it as a cron expression of the format '{second}{minute}{hour}{day}{month}{day of week}' to specify the schedule.

Note that many of the cron expressions you find online omit the {second} field, so if you copy from one of those you'll have to adjust for the extra field.

Here are some other schedule expression examples:

To trigger once every 5 minutes:

"schedule": "0 \*/5 \* \* \* \*"

To trigger once at the top of every hour:

"schedule": "0 0 \* \* \* \*",

To trigger once every two hours:

"schedule": "0 0 \*/2 \* \* \*",

To trigger once every hour from 9 AM to 5 PM:

"schedule": "0 0 9-17 \* \* \*",

To trigger At 9:30 AM every day:

"schedule": "0 30 9 \* \* \*",

To trigger At 9:30 AM every weekday:

"schedule": "0 30 9 \* \* 1-5",

If a specific .job file is not included in the directory, the following priority will be followed:

    1. Settings.job  
    2. Settings.env.job  
    3. CustomJobName.job  
    4. CustomJobName.env.job