Petroineos Development Challenge Requirements

Overview

The power traders require an intra-day report to give them their day ahead power position. The report should output the aggregated volume per hour to a CSV file.

Requirements

1. Must be implemented as a Windows service using .Net 4.5 & C#.
2. All trade positions must be aggregated per hour (local/wall clock time). Note that for a given day, the actual local start time of the day is ***23:00 (11 pm) on the previous day***. Local time is in the GMT time zone.
3. CSV output format must be two columns, Local Time (format 24 hour HH:MM e.g. 13:00) and Volume and the first row must be a header row.
4. CSV filename must be PowerPosition_YYYYMMDD_HHMM.csv where YYYYMMDD is year/month/day e.g. 20141220 for 20 Dec 2014 and HHMM is 24hr time hour and minutes e.g. 1837. The date and time are the local time of extract.
5. The location of the CSV file should be stored and read from the application configuration file.
6. An extract must run at a scheduled time interval; every X minutes where the actual interval X is stored in the application configuration file. This extract does not have to run exactly on the minute and can be within +/- 1 minute of the configured interval.
7. It is not acceptable to miss a scheduled extract.
8. An extract must run when the service first starts and then run at the interval specified as above.
9. It is acceptable for the service to only read the configuration when first starting and it does not have to dynamically update if the configuration file changes. It is sufficient to require a service restart when updating the configuration.
10. The service must provide adequate logging for production support to diagnose any issues.

Additional Notes

An assembly has been provided (PowerService.dll) that must be used to interface with the "trading system". A single interface is provided to retrieve power trades for a specified date. Two methods are provided, one is a synchronous implementation (`IEnumerable<PowerTrade> GetTrades(DateTime date);`) and the other is asynchronous (`Task<IEnumerable<PowerTrade>> GetTradesAsync(DateTime date);`). The implementation can use either of these methods. The class `PowerService` is the actual implementation of this service. The date argument should be the date to retrieve the power position (volume) for.

The `PowerTrade` class contains an array of `PowerPeriod`s for the given day. The period number starts at 1, which is the first period of the day and starts at 23:00 (11 pm) on the previous day.

The completed solution must include all source code and be able to be compiled from source. It may be delivered as a zip, zip of a DVCS (e.g. git) or a link to a hosted source control repository. This

should be emailed to ceri.lewis@petroineos.com (please also copy in ceri.lewis@gmail.com if you email a Zip file as these are often blocked by our company email filter).

Example

Given the call to PowerService.GetTrades returns the following two trade positions:

Date	Periods	
	Periods	Volume
01/04/2015	1	100
	2	100
	3	100
	4	100
	5	100
	6	100
	7	100
	8	100
	9	100
	10	100
	11	100
	12	100
	13	100
	14	100
	15	100
	16	100
	17	100
	18	100
	19	100
	20	100
	21	100
	22	100
	23	100
	24	100

Date	Periods	
	Periods	Volume
01/04/2015	1	50
	2	50
	3	50
	4	50
	5	50
	6	50
	7	50
	8	50
	9	50
	10	50
	11	50
	12	-20
	13	-20
	14	-20
	15	-20
	16	-20
	17	-20
	18	-20
	19	-20
	20	-20
	21	-20
	22	-20
	23	-20
	24	-20

The expected output would be:

Local Time	Volume
23:00	150
00:00	150
01:00	150
02:00	150
03:00	150
04:00	150
05:00	150
06:00	150
07:00	150
08:00	150
09:00	150
10:00	80
11:00	80
12:00	80
13:00	80
14:00	80
15:00	80
16:00	80
17:00	80
18:00	80
19:00	80
20:00	80
21:00	80
22:00	80

The expected output would be:

Local Time	Volume
23:00	150
00:00	150
01:00	150
02:00	150
03:00	150
04:00	150
05:00	150
06:00	150
07:00	150
08:00	150
09:00	150
10:00	80
11:00	80
12:00	80
13:00	80
14:00	80
15:00	80
16:00	80
17:00	80
18:00	80
19:00	80
20:00	80
21:00	80
22:00	80