

AWARE ELECTRONIC'S WINDOWS SERVICE APPLICATION AWARE_SERVICE.EXE

AWARE_SERVICE.EXE allows one to run Aware's Aw-Radw.Exe windows program as a Windows Service application. Windows Service applications are long-running applications that are ideal for use in server environments. Services can be automatically started when the computer is booted. They do not require a logged in user in order to execute and can run under the context of any user including the system.

Windows Services are controlled through the Service Control Manager where they can be stopped, paused, and started as needed.

Windows Services can be installed and controlled from a command line, using the Windows program SC.EXE. To learn more about SC.EXE use Windows help (enter SC in the Windows help screen).

When first starting, AWARE_SERVICE.EXE looks for a file named AWARE_SERVICE.SET containing lines of text telling AWARE_SERVICE.EXE what to do. It looks at each line in the AWARE_SERVICE.SET file, in turn, and executes the command.

Place the two files AWARE_SERVICE.EXE and AWARE_SERVICE.SET in the same folder (directory), for example C:\AWARE. Click on AWARE_SERVICE.SET to place it in Windows notepad, and edit it as below.

The first character in a command line is - (dash) followed by a command letter. The command letters can be either upper or lower case.

Following are the commands AWARE_SERVICE recognizes when reading the AWARE_SERVICE.SET file. *(Presented in order typically used):*

-R

Tells AWARE_SERVICE to rerun all the commands in the AWARE_SERVICE.SET file when the last program started with the -P command (see below) in the list exits. (For a use of the -R command, see below).

-Ddirectory

where directory is a path. This tells AWARE_SERVICE to switch to the directory path as the default directory.

For example:

-DC:\AWARE

tells AWARE_SERVICE to set the default directory to C:\AWARE

Without the -D command, AWARE_SERVICE will set the directory in which it resides as the default directory.

-M#

where # is 0, 1 or 2.

-M0 will cause AWARE_SERVICE to run any programs started with the -P or -C commands (see below) (not system commands) in a normal window. -M1 causes program(s) to run in a minimized window. -M2 cause program(s) to run in a hidden window.

Note to see any Windows programs started by AWARE_SERVICE, AWARE_SERVICE must be configured as interactive as in:

```
SC CONFIG AWARE_SERVICE TYPE= OWN TYPE= INTERACT START= AUTO
```

-S(milliseconds)

where (milliseconds) is a number. It causes AWARE_SERVICE to issue the Windows API Sleep command which tells Windows to cycle through all other program threads and then return to AWARE_SERVICE, after so many milliseconds. For example:

-S1000 would cause a Sleep delay of one second.

This can be used to allow previous command(s) to finish before the next line in AWARE-SERVICE.SET is executed.

-Isecs

(the letter i) where secs is a number. This command tells AWARE_SERVICE to wait until Windows has established a connection to the Internet. The number following the -I tells AWARE_SERVICE to wait for that many seconds before giving up waiting for Windows to established a connection to the Internet.

This can be useful if a command after the -I command depends on a connection to the internet for it to succeed, for example a batch file that uses ftp to copy files to an Internet site.

Example wait for internet connection but wait no longer than 20 seconds:

-I20

//comment

Tells AWARE_SERVICE to ignore anything on the line starting with //. This allows one to include comments in the AWARE_SERVICE.SET file.

-C followed by a command string.

Passes the command string to the command interpreter, which executes the string as an operating-system command. AWARE_SERVICE refers to the COMSPEC and PATH environment variables that locate the command-interpreter file (the file named CMD.EXE in Windows NT) and causes it to run the command.

With the -C command, AWARE_SERVICE will not wait for the command to finish before proceeding to the next command in the AWARE_SERVICE.SET file. One could include the -S command (see -S command above) after the -C command to cause AWARE_SERVICE to delay a certain amount of time before executing the next command in the AWARE_SERVICE.SET file

Typical -C commands include for example:

The COPY command:

-CCOPY C:\MY_DATA*.RAD C:\MY_OLD_DATA

Run a batch file and do not wait for it to finish:

-CC:\AWARE\MY_BATCH.BAT

If the path includes any spaces, include quotes as in:

-C"C:\MY AWARE\MY_BATCH.BAT" -ARG

For more information about system commands, search Windows Help for "Command-line reference A-Z" and "CMD".

-W followed by a command string.

The -W command is the same as the -C command except AWARE_SERVICE will wait for the command to finish before continuing with additional commands in the AWARE_SERVICE.SET file.

With the -W command, one should be sure the command will finish (exit) at some point in time. Otherwise AWARE_SERVICE will keep waiting for the command to finish although it can be stopped and restarted with Window's Service Control Manager or Window's SC.EXE program, or if start up type is set to AUTO, it will restart with a reboot.

-P followed by a program's name along with any arguments.

Start the program as a Windows Service. This normally would be the last command in the AWARE_SERVICE.SET file and the main program you want AWARE_SERVICE to start and watch.

As indicated, if the program is the main program you want AWARE_SERVICE to start and watch, it must be the last -P command in the AWARE_SERVICE.SET. It also should not be a batch file or script that launches another program because in such a case, AWARE_SERVICE will watch the batch file or script instead of the program launched by the batch file or script.

If the -P program exits and the -R command was not given, then AWARE_SERVICE will enter the stopped state. It then can be restarted with Window's Service Control Manager or Window's SC.EXE program, or if start up type is set to AUTO, it will restart with a reboot.

If the -R command (see above) is in the AWARE_SERVICE.SET file, and then if the program exits, AWARE_SERVICE will rerun all the commands in the AWARE_SERVICE.SET file.

Example -P command starting Aw-Radw:

```
-PC:\AWARE\AW-RADW -S -Lmy_data.rad
```

or

```
-P"C:\MY AWARE\AW-RADW" -S "-L\MY DIR\my_data.rad"
```

Notice the quotes around any program path or argument with spaces.

More about the use of the -R command

As indicated above, the -R command tells AWARE_SERVICE to rerun all the commands in the AWARE_SERVICE.SET file when the last program started with the -P command in the list exits.

Aware's program Aw-Radw responds to various command line arguments, as detailed in Aw-Radw's help file under the "Command-Line Arguments" heading. One of these command line arguments is the -Q arg.

From Aw-Radw's help file "Command-Line Arguments":

Aw-Radw -Qquit_after_points

The -Q argument followed by a number tells Aw-Radw to quit (exit) after the 'quit_after_points' number of radiation points have been gathered. For example the command line AW-RADW -AMY_FILE.TXT -Q1000 tells Aw-Radw to start collection of radiation data to an ASCII file named MY_FILE.TXT and to exit after 1000 points have been gathered. If Aw-Radw is started with the -Q arg., it will exit not only when the number of points have been gathered, but also if any error situation stops the collection of radiation data. For example if while gathering radiation data, if an error occurs, instead of just stopping radiation collection and displaying info. about the error, Aw-Radw will exit.

If the argument AW-RADW -Q-1 (minus one) is given, Aw-Radw will continue to gather radiation data with no limit on the number of points but will exit if an error occurs that stops the collection of radiation data.

Given the above information about Aw-Radw's -Q arg, one could create an AWARE_SERVICE.SET file containing:

```
-R  
-PC:\AWARE\AW-RADW -S -L -AMY_FILE.TXT -Q1000
```

When AWARE_SERVICE executes, it will start Aw-Radw. After 1000 points have been gathered or if some type of error causes Aw-Radw to stop collection (for example a temporary loss of communications with the radiation monitor by unplugging the USB port), AWARE_SERVICE will detect Aw-Radw has exited and then will rerun the commands in AWARE_SERVICE.SET, starting Aw-Radw again.

AWARE_SERVICE command line arguments

When AWARE_SERVICE starts, before it reads AWARE_SERVICE.SET, it looks at the command line passed to it for any command line arguments present and then acts upon them, as follows:

-L followed by a log file name as in:
-LC:\AWARE\AWARE_SERVICE.LOG

When AWARE_SERVICE is running, it will create a log file with a default name composed of the AWARE_SERVICE's file name before the .exe, followed by .TXT. In the case of AWARE_SERVICE.EXE the default log file name is AWARE_SERVICE.TXT. (One can change the name of AWARE_SERVICE.EXE to another name if desired, for example AW-SER.EXE in which case the default log file would be named AW-SER.TXT).

The log file will be written to the same directory where AWARE_SERVICE.EXE resides. AWARE_SERVICE.EXE places in the log file, the time and descriptions of the actions it carries out.

You can tell AWARE_SERVICE to use another log file with the -L command as in:

```
SC CREATE AWARE_SERVICE BINPATH= "C:\AWARE\AWARE_SERVICE.EXE -LC:\MYLOGS\AWARE_SERVICE.LOG"
```

-S followed by the set file path-name to use as in:

```
-SC:\AWARE\AW_SET.TXT
```

AWARE_SERVICE defaults to using a set file with the name AWARE_SERVICE.SET residing in the same folder that AWARE_SERVICE.EXE resides. To force it to use another set file use the command as in:

```
SC CREATE AWARE_SERVICE BINPATH= "C:\AWARE\AWARE_SERVICE.EXE -SC:\AWARE\SETUP.TXT"
```

-R

Same as -R command in the set file i.e. we want the last program started with the -P command, in the AWARE_SERVICE.SET list, to be watched, and if it exits, it will re-run all the commands in AWARE_SERVICE.SET.

The contents of an example AWARE_SERVICE.SET file

-R

```
//We want the last program in the list started with the -P command to be watched, and if it exits,  
//AWARE_SERVICE will re-run all the commands in AWARE_SERVICE.SET
```

-DC:\AWARE

```
//We want C:\AWARE to be the default directory (folder)
```

-CMYBATCH.BAT

```
//Have the system run mybatch.bat. You could include a path otherwise it will use the default directory as the  
path.
```

-S3000

```
//Sleep for three seconds to allow any commands in mybatch.bat to complete.
```

-PC:\AWARE\AW-RADW.EXE -LMYDATA.RAD -S -Q-1

```
//Start C:\AWARE\AW-RADW.EXE as a windows service
```

```
//with the three command line arguments -LMYDATA.RAD -S and -Q-1
```

```
//End of AWARE_SERVICE.SET
```

After AW-RADW.EXE starts, as per the above commands, AWARE_SERVICE will watch Aw-Radw.Exe and if it exits, AWARE_SERVICE will rerun all the above command

To install and start the AWARE_SERVICE

As mentioned above it is convenient to place AWARE_SERVICE.EXE and AWARE_SERVICE.SET in the same folder (directory), for example C:\AWARE. Click on AWARE_SERVICE.SET to place it in Windows notepad, and edit it as above.

See the included example batch files for typical applications.

You must be logged in as an administrator to install any Windows service. At the command prompt, from the RUN prompt, or from a batch file, execute:

```
SC CREATE AWARE_SERVICE BINPATH= "C:\AWARE\AWARE_SERVICE.EXE"
```

This runs the windows program SC.EXE telling it to create a service named AWARE_SERVICE using the exe at "C:\AWARE\AWARE_SERVICE.EXE". Notice the space after the binpath= . (Search Windows help for SC for help with SC.EXE)

Next execute:

```
SC CONFIG AWARE_SERVICE TYPE= OWN TYPE= INTERACT START= AUTO
```

This tells SC.EXE to configure AWARE_SERVICE as its own process (TYPE= OWN) and to allow it to interact with the desktop (TYPE= INTERACT) so you can see and interact with Aw-Radw's window after it starts as a service. The START= AUTO tells it to automatically start each time the computer is rebooted and run even if no one logs on to the computer. Notice the space after TYPE= and START= .

Next execute:

```
SC START AWARE_SERVICE
```

which tells SC.EXE to start the AWARE_SERVICE at which point AWARE_SERVICE starts and executes all the commands in the AWARE_SERVICE.SET file.

To stop the service:

```
SC STOP AWARE_SERVICE
```

To delete the service from Windows list of services:

```
SC DELETE AWARE_SERVICE
```

Note the service must be stopped before you delete it.

Once the service is created as in:

```
SC CREATE AWARE_SERVICE BINPATH= "C:\AWARE\AWARE_SERVICE.EXE"
```

one can use "Control Panel" – "Performance and Maintenance" – "Administrative Tools" – "Services" to configure AWARE_SERVICE in Windows SERVICE window, without using SC.EXE

After creating AWARE_SERVICE as above, Windows SERVICE window will list AWARE_SERVICE. Right click it then "Properties" - "General" and you can change "Startup Type" to automatic which will start AWARE_SERVICE

every time the PC re-boots. "Properties" - "Log On" with "Allow service to interact with desktop" checked, allows you to see and control the Aw-Radw window.

The contents of an example TEST.BAT file to create, config, start, stop and delete AWARE_SERVICE from Windows Service Control Manager

```
SC CREATE AWARE_SERVICE BINPATH= " C:\AWARE\ AWARE_SERVICE.EXE -L"  
echo Windows Service Control Manager should now list AWARE_SERVICE  
pause  
SC CONFIG AWARE_SERVICE TYPE= OWN TYPE= INTERACT START= AUTO  
echo Windows Service Control Manager should list AWARE_SERVICE with Startup Type Automatic  
echo and "Allow service to interact with desktop"  
pause  
SC START AWARE_SERVICE  
echo Aw-Radw should start and its window seen on the desktop  
pause  
SC STOP AWARE_SERVICE  
echo Aw-Radw should unload.  
pause  
SC DELETE AWARE_SERVICE  
echo Windows Service Control Manager should not list AWARE_SERVICE  
pause
```

If you have any questions, no matter how small, feel free to contact us.

Phone: 800-729-5397 or Phone-Fax: +302 655-3800

E-mail Aware@aw-el.com
[Aware Electronics Corp. Website](http://www.aw-el.com)
www.aw-el.com