

Alert when one is not Alert: SAS Call Sound Function

Vamse Goutam, Genpact, India

Amarnath Vijayarangan, Genpact, India

Palanisamy Mohan, Cognizant Technology Solutions, India

ABSTRACT

Multitasking is the order of the day and many SAS programmers work on multiple tasks simultaneously. For long running programs, the programmer assumes that this particular program might take certain amount of time, but in reality the program might get completed earlier or later than the expected time based on the load on server. The only way a programmer can check the completion of the program is by checking the log intermittently or by setting up an email alert, which can also be missed when the person is working offline. A prompt notification after the execution of the program can help and save lot of time for the programmer. The purpose of this paper is to leverage the functionality of Call Sound function which would help in effective time and resource management .

INTRODUCTION

SAS can alert the programmer through CALL SOUND function which alerts through a sound after a DATA step or a PROC step or program completion. The advantages of an alert function are

1. Programmer is notified even when he/she is not glued with SAS environment and is held up with other tasks.
2. Sometimes when there are multiple users working on a project, the programmer sitting adjacent to the current programmer can also hear the alert and can act upon it and submit next task assigned to him when the actual programmer is not around
3. When combined with Scheduler this can also serve the purpose of an Alarm

CALL SOUND SYNTAX

CALL SOUND (frequency, duration); (SAS help)

Frequency: specifies the sound frequency in terms of cycles per second. The frequency must be at least 20 and no greater than 20,000.

Duration: specifies the sound duration in milliseconds. The default is -1.

EXAMPLE1

The following statement produces a tone of frequency 523 cycles per second (middle C) lasting 2 seconds:

```
data _null_;  
  call sound(523,2000);  
run;
```

EXAMPLE2

The following example produces an alarm with regular breaks instead of a continuous single sound. This will produce sound for 5 times depending on the AlarmFrequency end value.

```
%macro callsound;  
  data _null_;  
    do AlarmFrequency=1 to 5;  
      do MakeAlarm=1 to 4;  
        call sound(700,700);  
      end;  
      call sleep(1000);  
    end;  
  run;  
%mend;  
%callsound;
```

EXAMPLE3

Calling the above callsound macro at the end of each program or any DATA / PROC step when the execution time is uncertain notifies the user about the task completion.

```
%macro test;
```

```
*** More SAS statements;  
*** More SAS statements;  
*** More SAS statements;
```

```
%mend test;
```

```
%test
```

```
%callsound;
```

EXAMPLE4

One can schedule an alarm in the system that could act as a remainder. In Windows environment, there are all kinds of third-party tools that allow you to set reminders for yourself in which the installation of these tools requires admin access to the system which may not be possible to every user. But as a SAS user let's explore how one can set alarm without using any third party tools but using SAS. SAS companied with Windows built-in Task Scheduler will be able to perform this task. Task scheduler is one of the system tools on Windows operating system where you can schedule multiple tasks and windows operating system will perform those tasks on a specified time and schedule.

For instance, if you want to set an alarm at 11 AM on specific days. Create the call sound macro in example2 and combine it with the task scheduler. However, to setup the task scheduler in Windows environment, you need to setup a batch file first. Please refer <http://www.pharmasug.org/proceedings/2011/AD/PharmaSUG-2011-AD11.pdf> for detailed information regarding how to schedule a job.

CONCLUSION

Call sound function is a classic example that exhibits the power of SAS in terms of till what extent one could use SAS for. This paper clearly states that SAS is just not for analytical and data management software but also could be used in our everyday life routine.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Vamse Goutam
Enterprise: Genpact
Address: Block B Salarpuria Soft Zone Outer Ring Road, Bellandur
City, State ZIP: Bangalore, Karnataka, 560103
E-mail: vamse.kobe09@gmail.com

Name: Amarnath Vijayarangan
Enterprise: Genpact
Address: Block B Salarpuria Soft Zone Outer Ring Road, Bellandur
City, State ZIP: Bangalore, Karnataka, 560103
E-mail: amarnath7@gmail.com

Name: Palanisamy Mohan
Enterprise: Cognizant Technology Solutions
Address: Block F2, Manyata Embassy Business Park, Outer Ring Road, Rachenahalli
City, State ZIP: Bangalore, Karnataka, 560045
E-mail: palanisas@gmail.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.