SAS Make

What is it?

SAS Make is a program for executing Windows programs sequentially. This is a natural thing to do in DOS programming, but not really available in Windows. The idea is to write a file and execute a line at a time.

Why do it?

I came up with the idea for this program while I was switching from DOS SAS to Windows SAS. I had several multi-file projects that I was controlling with MAKE. I needed something like MAKE to smooth the transition to Windows SAS.

Windows SAS has a batch mode capability, but I needed a method for sequentially executing files, and at the same time intermix calls to DOS. Although you can write a SAS program that executes all of these commands, there is no error recovery mechanism in the display manager for isolating errors. If your program has an error 1 hour into a 3 hour run, the rest of the program bombs. SAS Make lets you isolate the programs so that run-time errors are restricted to the current module.

How do you do it?

Write a program and execute it. There are some rules to follow for the statements, but they are simple enough that I can remember them. Note that this program is not restricted to executing SAS programs. You can issue any instructions as long as they are shorter than 1024 characters. Check out the syntax and options help.

Statement Syntax and Restrictions

Options

Run Time Behavior

How Do I PAY for this Wonderful Piece of Shareware?

If you like this program and use it regularly, please send \$25.00 US to

Mark Von Tress PO Box 171173 Arlington TX 76003

cis 71530,1170 email 71530.1170@compuserve.com

Components

This program is a Borland OWL 2 App Expert SDI Application. I used Strpp by Carl Moreland for strings and regular expression pattern matching. The wait routine is based loosely on execwait by Walter Oney's WinTech Journal article. The help file was developed using Visual Help by Firas Bushnaq. The setup program is SLS-Setup by Stephen Sama. Except for Borland's OWL, all of these are very good free or shareware programs you might like to look into.

Legaleze

SAS is a registered trademark of SAS Institute. DOS is a registered trademark of Microsoft Corporation Windows is a registered trademark of Microsoft Corporation

Disclaimer

DISCLAIMER: THIS PROGRAM IS PROVIDED AS IS, WITHOUT ANY WARRANTY, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO FITNESS FOR A PARTICULAR PURPOSE. THE AUTHOR DISCLAIMS ALL LIABILITY FOR DIRECT OR CONSEQUENTIAL DAMAGES RESULTING FROM USE OF THIS PROGRAM.

Syntax

Each line of the file is a system command, just as if you were to issue it from the Run menu item from the Program Manager file menu. There are some special directives that simplify executing instructions.

Directives

Directives are lines beginning with special strings.

n Lines beginning with "//" are comments and are not executed. Here is an example

//this command won't be executed because it is a comment

n Lines beginning with "{dos}" are DOS programs or commands internal to command.com. If you want to redirect a directory listing from the DOS dir command, you need to put the following line:

{dos} dir *.* > junk.out

The {dos} directive must be used for internal DOS command such as dir, copy, del, type, etc. This directive calls the default command interpreter with the /c option. The standard DOS command line is

command.com /c dir *.* > junk.out

In Windows, this command starts _default.pif with the command you want to execute. Standard DOS executables do not need the {dos} directive because Windows knows how to run them. Also note that the {dos} directive is needed when you use dos commands that use redirection or piping, ie symbols such as "<" ">" ">" or "|".

You should make sure that _default.pif has the "Close Window on Exit" check box checked, and allow background exection. Use the pif editor to set these options.

You can reconfigure the {dos} directive in the options dialog box

n Lines beginning with "{sas}" execute the default executable and command line options in the options dialog box. The following two lines are equivalent

{sas} myfile.sas
c:\winsas\sas.exe -config c:\winsas\config.sas -sysin myfile.sas

You can substitute whatever automatic command you want the {sas} directive to be in the options dialog box. You can also reconfigure the command line options for the {sas} directive.

- n Lines beginning with "{exit}" stop the program.
- n Lines beginnning with " {message}" send a message box to the screen. You need to clear the message box before the program can continue. The text following {message} will be displayed in the message box. Here is an example:

{message} This text will appear in a message box!

n Lines beginning with "{beep}" make a beep on the speaker. You can use this to notfiy you that the job has finished.

Restrictions

- n Lines can be at most 1023 characters long, and must be on a single line.
- n Blank lines are ignored.
- n Files are restricted to be 64K by the editor. The editor is simialar to notepad, and there is only one editor available. Hence you cannot merge two files. It can accept text input from the clipboard, so you can load text into notepad, and cut and paste it into sasmake. As far as editors go, it is slightly better than notepad.
- n It doesn't do graphics.
- n Commands are converted to lower case before execution.
- n This program requires Windows 3.1, and that you have installed toolhelp.dll with Windows. Toolhelp.dll is usually in your Windows system directory. I have only tested it in enhanced mode, so I bet enhanced mode is required since multitasking must be available.

Options Dialog Box

The options dialog box is accessed from the File menu. The options are:

- n Initial Directory: This is the directory where your first file may be opened. It is the same as the last directory you worked in. It is saved in sasmake.ini when you exit sasmake.
- n Default path and executable file name: this is the default command that is executed for the {sas} directives
- n Default command line options: this is the command line option for the {sas} directive.
- n Wait for termination: You don't have to wait for a process to terminate if you don't want to. If this check box is turned off, all the commands in your file will spew forth as fast as they can. It is useful if you want to start a bunch of Windows apps at once.

This is a dangerous option, and I accept no responsibility for what happens. You can have several applications trying to write to same file at once, and it isn't clear what will happen.

n Default DOS command interpreter: This is the command interpreter for executing DOS commands. This is the command issued by the {dos} directive. It loads command.com with the /c option. In DOS, this would load a copy of command.com and execute the command following the /c options, and then unload command.com. In Windows, this calls _default.pif and executes the command.

Saving the Options in SASMake.ini

The options are stored in a Windows initialization file called sasmake.ini. It can be found in your Windows Directory. Your current options settings are saved when you exit SASMake. You can restore the defaults using the restore defaults button in the dialog box

Run Time Behavior

File Handling.

Open a file to execute. The first open file dialog will start in the last directory where you were using SAS Make. I like to use the extension .msv for SAS Make files. Then you can associate SAS make to this extension in the Windows File Manager. Edit the file. Then save it.

Execution

There are several ways to run the program. Select the run option from the file menu. You can start the program using the alt-r or alt-R key combinations. You can start the program by selecting the run button on the speed bar.

The program will minimize itself to an icon, and disable keyboard or mouse input. It will start each of the files in your submitted file. SAS Make waits for Windows to send a message to it, and checks if 0.5 seconds have passed. If 0.5 seconds have passed, then it checks if the program is still running. If it is still running, it yields control to other applications. If the application has stopped, it moves to the next application.

When it has finished, SAS Make Restores itself, and allows input messages again.

Error Reporting

- n A message box will appear if an command could not be executed. This error usually occurs when you issue an internal DOS command without the {dos} directive. It can also happen if you have exhausted your Windows resources. You will be notified of the error and given the opportunity to stop the program.
- n SAS Make moves to the next line after you clear the error message box if you choose to continue.

HELP

The help system may be accessed several ways. You can use the help menu item "syntax and running". You can press the F1 key. You can press the Alt-h or Alt-H key combinations.