

SuperSet™ 1.1 - DOS Environment Extender Tool

User Manual

Copyright © 1992-1995 by Ross Leidy

TABLE OF CONTENTS

μWHAT IS SUPERSET?.....	3
WHO NEEDS SUPERSET?.....	3
SUPERSET FILES.....	3
QUICK START.....	4
SUPERSET FEATURE DETAILS.....	5
/A and /P (Append and Prepend).....	6
/S , /U and /L (Save, Update and Load).....	6
/D (Delete specified environment variable).....	7
/K (Delete the entire environment).....	8
/E and /N (Manipulation of Novell NetWare drives in the PATH).....	8
/M (Use the master environment).....	8
/R[:nnn] (Reserves nnn bytes of environment space, 256 by default).....	9
/Z (Size information.).....	9
/C (Compressed mode. Displays environment like DOS 'SET').....	9
/Q (Quiet mode. No confirmations).....	9
? or /H (Help screen).....	10
TROUBLESHOOTING.....	10
SHAREWARE INFORMATION.....	11
CONTACTING THE AUTHOR.....	12
WHERE TO FIND SUPERSET.....	13
VERSION HISTORY.....	13
ACKNOWLEDGMENTS.....	13
DISTRIBUTION.....	14
COPYRIGHT.....	14
TRADEMARK REFERENCES.....	14
DISCLAIMER.....	14

WHAT IS SUPERSET?

SuperSet™ is a shareware program that replaces the DOS 'SET' command. Like the DOS 'SET' command, SuperSet allows you to view and modify DOS environment strings. SuperSet, however, contains additional features that make it a superior tool. With SuperSet you can:

- Extend the length of any environment variable to any arbitrary length. Beat the normal DOS limit of 127 characters.
- Prepend or append to existing environment variables. No more re-typing your entire path just to add one more directory.
- Save, Load and Update environment snapshot files. Quick, one-shot way of setting-up your environment in one command.
- Replace or delete environment variables.
- Modify the parent OR the master environment block.
- Prune Novell NetWare network drives in the PATH.
- Display all environment strings in an easy-to-read format.
- Display environment size and usage information.

WHO NEEDS SUPERSET?

SuperSet contains features that make it useful to:

- Software developers who have a surplus of development tools on their computers. No more choosing which directories to exclude from the PATH. Using SuperSet, now all your tools can be accessible through the PATH.
- Network Administrators who are responsible for maintaining multiple, identically-configured computers can use the snapshot feature to setup environments. When a network-based snapshot file is used, changes to that file can effect changes to the PATH of multiple computers.
- Anyone who is frustrated by the limited control provided by the DOS 'SET' command.

SUPERSET FILES

Because of varying software distribution methods, the archived filename of SuperSet can vary. The archived filename will be one of SSET11.ZIP, SSET11.ARJ, SSET11.SDN, etc. Regardless of the filename of your copy of the archive, it will expand into the following files:

SUPERSET.EXE	- SuperSet program
SUPERSET.DOC	- Manual in Word for Windows 2.0 format
SUPERSET.TXT	- Manual in ASCII text format
SUPERSET.REG	- Registration form in ASCII text format
README.TXT	- Last minute information
FILE_ID.DIZ	- BBS description file
SDN.ID	- Software Distribution Network info

QUICK START

Those who are familiar with the DOS 'SET' command will feel right at home with SuperSet. In its simplest form it matches the syntax of the DOS 'SET' command. So, for example, just typing:

```
SuperSet
```

displays all of the environment strings. Or, typing

```
SuperSet path=c:\dos;c:\borlandc\bin;
```

sets the PATH variable. To add to the PATH variable without retyping the existing string, use the '/A' switch to append or the '/P' to prepend. For example,

```
SuperSet /A path=c:\123;c:\excel;
```

modifies the PATH variable so that it now has the value:

```
c:\dos;c:\borlandc\bin;c:\123;c:\excel;
```

SuperSet allows you to save and restore environment snapshots. With a snapshot you can restore the entire environment with one command. To save a snapshot of the current environment, type:

```
SuperSet /S
```

By default, this saves the contents of the environment in the file 'C:\SUPERSET.ENV'. Later, when you want to restore the environment, simply type:

```
SuperSet /L
```

One deviation from the semantics of the 'SET' command is the meaning of commands like:

```
SuperSet path=
```

In this case, SuperSet will display the current value of the PATH variable. A similar DOS 'SET' command would have just deleted the PATH variable. SuperSet will not delete an environment variable without the '/D' switch on the command line. So, the command to delete the PATH variable is:

```
SuperSet /D path
```

SuperSet sets the DOS ERRORLEVEL if an error occurs during processing. This allows you to test the success of the specified operation from within batch files.

With this brief introduction, you should be able to get started using SuperSet. The next section details the powerful features of SuperSet. If you prefer the hands-on approach and want to start using it right away, then you can refer to the SuperSet help screen that briefly

describes the available features — just type:
SuperSet /?

SUPERSET FEATURE DETAILS

The syntax of SuperSet is:

```
SuperSet [switches] [variable [=] [value] ]
```

The [switches] are:

- /A - Append to the existing variable value
- /P - Prepend to the existing variable value
- /S - Save an environment snapshot to file
- /U - Update an environment snapshot file
- /L - Load an environment snapshot from file
- /D - Delete specified environment variable
- /K - Delete (kill) the entire environment
- /E - Move network directories to the end of the PATH variable
- /N - Remove network directories from the PATH variable
- /M - Use the master instead of the parent environment
- /R[:nnn] - Reserves nnn bytes of environment space
- /Z - Displays environment size information
- /C - Compressed mode. Displays environment like DOS 'SET'.
- /Q - Quiet mode
- /? /H - Help and registration information.

All switches are case-insensitive (e.g., '/S' is equivalent to '/s').

The [variable] is the environment variable name up to the '='. SuperSet will not allow spaces or tabs to occur within the environment variable name. (This is different than the DOS 'SET' command. See the Troubleshooting section for a discussion of this difference.)

The [value] is everything that follows the '='. All characters after the '=' are preserved as-is when setting the environment string.

The '=' that separates the [variable] from the [value] is optional. This provides an abbreviated method of specifying environment strings while still allowing the use of the 'SET' syntax. Without a '=', SuperSet considers the first non-whitespace character after [variable] to be the start of [value]. For consistency, all examples in this manual have the '=' specified.

Most switches will work in combination with others. In the following descriptions, the switches that are grouped together (e.g., /A and /P) are mutually exclusive; only one of the group may be used on the command line at one time. If more than one switch within a group is specified, only last one on the command line will be processed. The others will be ignored.

SuperSet will display the contents of the environment block when invoked with no arguments. The addition of "passive" switches (/Q, /C, and /M) to the command line alters the display slightly, but SuperSet still displays the entire environment block. When using any of the other switches or arguments, SuperSet will display text that reflects the result of the command.

/A and /P (Append and Prepend)

These switches are used to modify values of existing environment variables by appending (/A) or prepending (/P) strings. These switches provide two major benefits over the DOS 'SET' command.

- 1) You don't have to type out the entire current environment string if all you want to do is add to it.
- 2) You can create environment variables longer than 127 characters.

Multiple invocations of SuperSet using these switches allow you to create a PATH of any length up to the size of your environment block. (Of course, you can extend any environment variable's length, but the PATH variable is one of the most useful)

Examples:

```
SuperSet PATH =C:\DOS;C:\UTILS;C:\EPSILON;C:\WINDOWS;
SuperSet /A PATH =C:\TOOLS\WATC\BIN;C:\BORC31\BIN;
:
:
```

Notice the spacing in the preceding examples. SuperSet disregards any whitespace (space or tab) that occurs before the equal sign. Because of this behavior, SuperSet modifies the same variable whether it appears as "PATH =" or "PATH=" on the command line. Conversely, all of the characters that follow the equal sign are preserved. Therefore, in these examples, no spaces appear after the equal sign.

/S , /U and /L (Save, Update and Load)

These switches are used to save (/S), update (/U) and load (/L) an environment snapshot file. A snapshot is useful after you have configured your environment and it's just right. You can save a snapshot and later restore it to the same exact state. For example, you can setup your entire environment inside AUTOEXEC.BAT with one command!

All of the snapshot switches reference the 'SUPERSET' environment variable and use the value as the path and filename for the snapshot file. If the variable 'SUPERSET' isn't defined (probably most cases), then SuperSet defaults to the file 'C:\SUPERSET.ENV'.

The contents of snapshot files may be edited and changed. Just be sure to follow these two guidelines when making any modifications to a snapshot file:

- 1) You can add comment lines by placing a '#' in the first column. The remainder of the line will be ignored.
- 2) Each variable=value combination must occupy only a single line in the file, no matter how many characters are in the line.

When creating a snapshot file (using '/S'), SuperSet will save the entire contents of the environment. The generated file will contain a comment block that indicates that it is a SuperSet snapshot file followed by all of the environment strings. If the save operation would overwrite an existing file, SuperSet will prompt you before

overwriting the file. The '/Q' (quiet) switch will override the prompting and will unconditionally overwrite the file. This is useful when running SuperSet from within batch files.

When updating a snapshot file (using '/U'), SuperSet will scan the contents of the existing snapshot file and save all the current environment values of the strings it finds. If a variable appears in the snapshot but is not in the current environment, the old value is used in the new snapshot. This feature is convenient when you've just added a new subdirectory to your PATH (using SuperSet, of course) and you want to save that change in a customized snapshot file. When loading a snapshot file (using '/L'), SuperSet will overwrite variables of the same name that already exist in your environment. Variables which exist in your environment, but not in the snapshot file will be left untouched. If you wish to start with a clean slate before loading in a snapshot, you can use the '/K' switch in conjunction with the load command (refer to the '/K' section for more details).

Examples:

```
(If the environment variable 'SUPERSET' doesn't exist)
SuperSet /S   (Saves snapshot in C:\SUPERSET.ENV)
SuperSet /P PATH=C:\4DOS; (Prepend to the PATH)
SuperSet /U   (Updates snapshot in C:\SUPERSET.ENV)
:
:
SuperSet /L   (Loads snapshot from C:\SUPERSET.ENV)
```

```
(Define the variable 'SUPERSET' and save a snapshot. SuperSet will define the
environment variable first and then perform the save operation.)
SuperSet /S SUPERSET=D:\ENV\SNAPSHOT.1
```

```
(Now that the variable 'SUPERSET' is defined)
SuperSet /L   (Loads snapshot from D:\ENV\SNAPSHOT.1)
```

/D (Delete specified environment variable)

SuperSet will not delete any environment variables without this switch. Furthermore, if a variable/value combination is entered, the delete will not be processed. The variable must appear by itself before the variable will be deleted.

Examples:

```
SuperSet /D PATH=C:\DOS; (Nothing deleted)
SuperSet /D PATH=       (PATH deleted)
SuperSet /D PATH       (PATH deleted)
```

/K (Delete the entire environment)

This switch may be useful when loading an environment snapshot file if you want to purge the environment of unwanted strings. It completely eliminates all environment contents. No matter

where this switch appears among other switches, the environment clearing will occur first.

Examples:

(The following are equivalent. Both delete the entire environment and then load the snapshot file.)

SuperSet /K /L

SuperSet /L /K

/E and /N (Manipulation of Novell NetWare drives in the PATH)

In a Novell NetWare environment, logging into a file server often results in some network drives being mapped and prepended to the PATH. I found that programs that reside on my local hard disk were often duplicated somewhere on a network drive. When I ran a program, DOS would search the network drives first and find the program there. Even when the program was not located on the network drives, there was additional search time introduced for every command I typed because of the search order. SuperSet helps me get around this problem. The '/E' switch will scan the PATH variable and look for drive entries of the form:

<d>:. (drive letter, colon, period)

Any matching entries found in the PATH are migrated to the very end of the PATH. At the end of the PATH, network drives won't interfere with the normal operation of the DOS PATH but they will still be accessible.

In addition, I found that sometimes I wanted to purge my path of the network drives. The '/N' switch will remove all entries in the PATH that match the above pattern.

Examples:

(Initially, PATH=X:.;Y:.;Z:.;C:\DOS;)

SuperSet /E

(Now, PATH=C:\DOS;X:.;Y:.;Z:.)

SuperSet /N

(Now, PATH=C:\DOS;)

/M (Use the master environment)

By default, SuperSet operates on the environment block of the parent application. The '/M' switch forces the use of the master environment block (the one belonging to the command processor that was last invoked). This is useful if modifying the environment from inside makefiles. Normally, SuperSet is invoked from a batch file, and, in that case, the Master and the Parent environments are one and the same.

Examples:

(Modify the TEMP variable of the master environment)

SuperSet /M TEMP=E:\TEMPDIR

/R[:nnn] (Reserves nnn bytes of environment space, 256 by default)

This is definitely the most esoteric feature of SuperSet. I once found it necessary to invoke SuperSet from within another program (not a batch file) and have SuperSet modify the

environment block of that program. This would not be a problem except for the fact that DOS, when preparing to load programs, makes a copy of its own environment block for use by the program and shrinks the block down to eliminate most unused space. So, no matter how much extra space may be available in DOS' environment block, that extra space is purged from the copy. Because of this behavior, there is very little room in the environment block to add variables. By first invoking SuperSet with the '/R' switch at the DOS prompt, SuperSet creates a special environment variable in DOS' environment block. Now, when invoking SuperSet from within another program, that special variable gets copied, along with all other variables, into the environment copy. This provides extra space for SuperSet to use for creating variables from within other programs.

You really have to go out of your way to come up with a situation where this feature is required. Modifying the environment block of running programs is a tricky affair. It assumes that the program will refer back to its environment block after it has begun running; it often will not. Examples:

```
SuperSet /R           (Reserves 256 bytes by default)
SuperSet /R:512      (Reserves 512 bytes)
```

/Z (Size information.)

This switch provides some additional information about the state of the environment. SuperSet reports the environment block's location, total block size, used size, and the amount free. It's good to check your environment usage every once in a while to make sure you're not too close to the edge. When you've got very little left, it may be time to increase your environment block. (See the Troubleshooting section for instructions on how to do this.)

Example:

```
SuperSet /Z           (Displays environment size info)
```

/C (Compressed mode. Displays environment like DOS 'SET')

By default, SuperSet columnizes the display when showing the environment strings. If you want a compacted form similar to the DOS 'SET' command, use the '/C' switch.

/Q (Quiet mode. No confirmations)

Normally, SuperSet displays a brief header and confirmation text that lets you know the results of your command. Sometimes, however, you may just want SuperSet to perform the action without displaying any information. This is particularly true when calling SuperSet from within a batch file (e.g., AUTOEXEC.BAT). The '/Q' switch disables the display of this information. In addition, certain commands may ask for user confirmation (e.g., the Save Snapshot switch). That behavior can sometimes be irritating if invoked from within a batch file, so this switch will bypass those confirmations.

Error messages are never suppressed.

/? or /H (Help screen)

These switches present a brief description of the commands that are available and how to register SuperSet. If either of these switches appear on the command line for SuperSet, all other switches and arguments will be ignored.

TROUBLESHOOTING

One of the major uses of SuperSet is its ability to extend environment variables to an indefinite length. In order to be able to do this, the master environment block must have enough unused space. The master environment block is created by the command line interpreter (COMMAND.COM, 4DOS.COM, etc.) when it is started. By default, COMMAND.COM creates an environment block that is fairly small (256 bytes). In order to take full advantage of SuperSet's capabilities, you may need to increase this size. Fortunately, it is very easy to do. Your CONFIG.SYS file should have a 'SHELL' statement that specifies which command line processor to start. It will look something like this:

```
SHELL = C:\DOS\COMMAND.COM
```

If your file doesn't contain a SHELL line you can add one. The SHELL statement simply tells DOS what command line interpreter to use and where it is located. If you don't have a SHELL statement, DOS will behave as if your SHELL statement is this:

```
SHELL = C:\COMMAND.COM
```

Just add a line like that to the top of your CONFIG.SYS file.

If you already had a SHELL statement, the line may contain some additional arguments, too. To increase the environment size, simply add a '/E:nnn' switch to the end of the 'SHELL' line (or modify if it's already there). For example, if you need a 1024 byte environment block, the 'SHELL' line should now look like this:

```
SHELL = C:\DOS\COMMAND.COM /E:1024
```

The next time you reboot, COMMAND.COM will create a 1024 byte environment block. You can type 'SuperSet /Z' to check the size of the block and the environment usage.

Whenever you define variables longer than 127 characters, and you use the DOS 'SET' command to display the environment contents, it will probably not display everything properly. Don't worry, your environment is intact. It is simply a deficiency in 'SET'; it doesn't expect to encounter those long variables. Just use SuperSet with no arguments to see the contents of your environment.

Registration is also available on-line through CompuServe. To register SuperSet via CompuServe, login and then simply:

- Type 'GO SWREG'. DOSCIM and WINCIM users can select 'GO' from the 'Services' menu or from the button-bar.
- Follow the directions for registration. When asked for SuperSet's Registration Database ID, enter **4556**.

Regardless of the method you choose, your registration will be personally and promptly acknowledged. I'll also let you know of any updates that are available, any bugs that I'm aware of and fixing, and any new products on the horizon. Site licenses are available at a discount. Please contact the author for arrangements.

This program is produced by a member of the Association of Shareware Professionals (ASP). ASP wants to make sure that the shareware principle works for you. If you are unable to resolve a shareware-related problem with an ASP member by contacting the member directly, ASP may be able to help. The ASP Ombudsman can help you resolve a dispute or problem with an ASP member, but does not provide technical support for members' products. Please write to the ASP Ombudsman at 545 Grover Road, Muskegon, MI 49442-9427 USA, FAX 616-788-2765 or send a CompuServe message via CompuServe Mail to ASP Ombudsman 70007,3536.

CONTACTING THE AUTHOR

If you experience any problems with SuperSet, you can send me email at:

From the internet:
75122.1275@compuserve.com

From CompuServe:
75122,1275

You can also send postal mail too:

Ross Leidy
1966 Staunton Road
Cleveland Heights, OH 44118
USA

I do my best to fix all reported bugs as quickly as possible. Unregistered users are welcome to send in bug reports. If I can fix those bugs, I may be able to win over a few more registered users. Bug reports should contain the version of SuperSet, version of DOS, the nature of the bug, and any other pertinent information.

Also, anyone should feel free to drop me email if you have any questions or ideas for enhancements.

WHERE TO FIND SUPERSET

The best places to find the most current version of SuperSet are:

- On the internet in the SimTel Software Repository and its mirror sites such as oak.oakland.edu. The specific directory is SimTel/msdos/envutil.
- On CompuServe in the PCUTIL forum, DOS/DOS Utils library.
- On any BBS that receives files from the Shareware Distribution Network (SDN) Project.

Look for a file in the form:

SSETxx.ZIP

where the xx is the major/minor version number. For example, SuperSet v1.1 would appear as:

SSET11.ZIP

Although you may have found SuperSet at a different internet site or BBS, all new versions of SuperSet will be posted to these sites first.

VERSION HISTORY

- 1.1 - Initial public release.
 - Added environment snapshot support.
 - Added '/M' for master environment modification.
 - Added '/C' to mimic DOS 'SET' display of environment.
 - Added '/K' for environment deletion.
 - Improved help screen.
 - Cleaned up a few cosmetic bugs.
- 1.0 - Initial release. This version underwent multiple revisions while being used by the software development team of a computer peripherals company. This version was not released to the general public.

ACKNOWLEDGMENTS

Many thanks to those who have registered SuperSet and are keeping the Shareware concept a viable one. Thanks to those people who participated in the initial testing of pre-released SuperSet. Special thanks to:

- Ted Milbaugh of Green Light Software. Thanks for the ideas for the user manual.
- My wife Renee for support, patience, understanding, and great scones.

DISTRIBUTION

SuperSet may be copied and distributed freely, provided that all the original files are included. All code and documentation must remain intact and unmodified. SuperSet may be included in Shareware CD-ROMs or other Shareware collections, provided that acquiring the collection does not remove the user's obligation to register the software.

COPYRIGHT

This software is Copyright ©1992-1995 by Ross Leidy, all rights reserved. SuperSet is a trademark of Ross Leidy.

TRADEMARK REFERENCES

MS-DOS and Windows are registered trademarks of Microsoft Corp.

CompuServe is a registered trademark of CompuServe, Inc.

Novell and NetWare are registered trademarks of Novell, Inc.

DISCLAIMER

THE AUTHOR DISCLAIMS ALL WARRANTIES RELATING TO THIS SOFTWARE, WHETHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND ALL SUCH WARRANTIES ARE EXPRESSLY AND SPECIFICALLY DISCLAIMED. NEITHER THE AUTHOR NOR ANYONE ELSE WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION, OR DELIVERY OF THIS SOFTWARE SHALL BE LIABLE FOR ANY INDIRECT, CONSEQUENTIAL, OR INCIDENTAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE SUCH SOFTWARE EVEN IF THE AUTHOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR CLAIMS. IN NO EVENT SHALL THE AUTHOR'S LIABILITY FOR ANY DAMAGES EVER EXCEED THE PRICE PAID FOR THE LICENSE TO USE THE SOFTWARE, REGARDLESS OF THE FORM OF THE CLAIM. THE PERSON USING THE SOFTWARE BEARS ALL RISK AS TO THE QUALITY AND PERFORMANCE OF THE SOFTWARE.

Some states do not allow the exclusion of the limit of liability for consequential or incidental damages, so the above limitation may not apply to you.

This agreement shall be governed by the laws of the State of Ohio and shall inure to the benefit of Ross Leidy and any successors, administrators, heirs and assigns. Any action or proceeding brought by either party against the other arising out of or related to this agreement shall be brought only in a STATE or FEDERAL COURT of competent jurisdiction located in Cuyahoga County, Ohio. The parties hereby consent to in personam jurisdiction of said courts.