

MTWIN 2.06

Mtwin is a small Utility to modify any kind of INI File.

Usage:

Mtwin uses Command files (Extension CMD), which contain the commands for modifications that should be made to a given INI file.

Usage:

```
mtwin [options] <ICM-file> [options]
mtwin [options] <INI-file> [options] <CMD-file> [options]
mtwin [options] <GRP-file> [options] <GRI-file> [options]
mtwin [options] <GRP-file> [options] <CMD-file> [options]
```

```
Options:  starts with - or /
/b       Create Backupfile
/c       Compile GRI into GRP file
         When only one filename is specified,
         a ICM file is used
/d       Decompile GRP into GRI file
/g       GRP & CMD file
/h /?    Display this help-screen
/l=<file> Log messages to file
/n       Creates INI files, if they do no exist
/o       Overwrite same file
/s       Swap GRP-Header
/v[0-9]  Verbose mode, default value is /v3
         /v0 Display no messages
         /v9 Display all messages
```

If you want to modify multiple INI files, you can create an ICM (Ini Command File). Each line of this file contains an INI file filename and a Command file filename.

```
;  
; Example ICM File  
;  
C:\WINDOWS\WIN.INI      C:\UT\WIN.CMD  
C:\WINDOWS\SYSTEM.INI  C:\UT\SYSTEM.CMD
```

A backup of the INI-File with the extension .BAK is made before modifying the file. If the /b parameter is specified, the backup file is not deleted after processing.

When decompiling GRP files, normaly the Icon-name (as displayed in the group) is used as the section-name. When your users change the Iconnames, then you can use the /s option, to specify that the EXE-Name is used as section-name, instead of the Iconname. Of course, when recompiling the GRI file, you must use the same option.

All lines which begin with ; are treated as comments.

Normally all section names, keys and values are NOT case sensitive. However, _ and * give the possibility for case sensitive matching.

MTWIN can process INI files of any size, even larger than 64 kBytes. However, a dynamic list of all commands within a CMD file is created in memory as it is loaded from the command line or an ICM file and could reduce available memory temporarily. This is normally not a problem, as you can use more than ??? commands in one CMD file. If this isn't enough, you can split the CMD file and use an ICM file to call the separate CMD files individually.

INI Commands

;	When this is in front of a line the line is ignored.
-[section]	The corresponding section is deleted
-[section]Key	The Key in section will be deleted
-[section]Key=Value	The Key is only deleted if the Value matches exactly This is used to delete device=... lines in system.ini
_[section]Key=Value	Same as - but Value is case sensitive Normally not used.
~[section]Key=Value	Used to delete Keys, when the Key is terminated by a numeric value. (Used to delete Groups in PROGMAN.INI)
#[section]	The corresponding section is commented
#[section]Key	The Key in section will be commented
#[section]Key=Value	The Key is only commented if the Value matches exactly This is used to comment device=... lines in system.ini
@[section]	The ; in front of the corresponding section is removed
@[section]Key	The ; in front of the Key in section is removed
@[section]Key=Value	The ; in front of the Key is removed, if the Value matches exactly. This is used to uncomment device=... lines in system.ini
+[Section]Key=	Adds Key in Section.
+[Section]Key=Value	Adds Key and Value to section.
?+[Section]Key=Value	Adds Key and Value to section, but ONLY if the key does not currently exist.
&[Section]Key=Value	This Key/Value is always inserted, even if already a key exists, but with a different value. Used to insert device=... lines in System.ini
*[Section]Key=Value	Same as &, but the value is case sensitive Normally not used.
![Section]Key=Value	A key is inserted in section, but a numeric value is added to Key, representing the highest+1 value of Key. (Only ?) used for Progman.ini to add new program groups. ~ is used to delete such keys if needed.

If the value of the command is already found, then the line is not duplicated. You can add multiple ! entries per Section/Key pair in one pass

>[Section]Key=Value	<p>Adds Value to an existing value. The old Value is preserved and the Value is added at the end of the line.</p> <p>When the values are separated with a coma, (like network= in system.ini) then you must add a coma sign before the value. Like >[386enh]networks=,vipx.386. The coma sign is then inserted when used.</p>
/[Section]Key=Value	<p>Same as >, but the Value is inserted in front of all existing values.</p> <p>When the values are separated with a coma, (like network= in system.ini) then you must add a coma sign after the value. Like >[386enh]networks=vipx.386,. The coma sign is then inserted when used.</p>
<[Section]Key=Value	<p>Deletes Value from the Key/Value line. Coma signs should be handled correctly.</p>
r[Section]Original=NewValue	<p>The string <Original> is replaced with <NewValue>, but only in if [Section] is the current one. If you want to replace the Original in ALL sections (even in the section definition) then specify [] as the section. When the original value include a = character, then you must enclose it with ".....". When the " character is used as single character, then you must add a \ in front of it.</p>
R[Section]Original=NewValue	<p>Same as r command, but the case must match.</p>
S[Section]Key=<EnvVarname>	<p>The environment variable with name <EnvVarname> is set to the Value of this Section/Key.</p> <p>If [Section]Key does not exists, the environment is not modified.</p> <p>The modification is done in the master environment, which is the one of the FIRST command.com loaded.</p> <p>If you use this command under windows, the environment of the shell inside windows is NOT modified, but when you leave windows, the environment has the modified values !</p>

Conditional execution (New Style)

You can use structured IF-commands. They have the following form:

```
IF <condition>
..
..
..
ELSE
..
..
..
ENDIF
```

The IF's can't be nested. The following conditions are currently implemented:

VIDEO == != Test if the videoadapter is one of the following

- NODISPLAY
- UNKNOWN
- MDPA
- CGA
- HERCULES
- MCGA
- EGA
- VGA
- SVGA

MONITOR == != Test if the monitor is one of the following

- MONO
- COLOR
- ENHANCED
- ANALOGMONO
- ANALOGCOLOR

MEMEBER OF <value> Test if the user is member of the given group. Works currently only with Netware 2.x, 3.x and 4.x with bindery emulation. NDS support is planned, but due to missing support from Novell not to be expected soon. (Sorry) There is a small **utility called ismember.exe**, who returns errorlevel = 0 or 1, depending if the user is in the given group.

ENV <envvariable> == != <value> Test if the environment variable has the given value. When the <value> has blanks inside, then you must use a leading " and a terminating " character.

EXIST FILE <filename>
EXIST DIR <dirname>
EXIST ENV <value> Test if the given file/directory/environment value exists.

NEXIST FILE <filename>
NEXIST DIR <dirname>
NEXIST ENV <value>

VESA == <value>
VESA != <value> Test for presence of a VESA-Bios (or extension-driver). You can use the included utility **vesatest.exe** to display the string returned by VESA. For the Compaq Advanced VGA adapter it is : Compaq Adv VGA Ext (512kB Ram) So the line would be:
 IF VESA == "Compaq Adv VGA Ext (512kB RAM)"

INFILE <disk-file> SECTION <value> Tests for the presence of the given section in <disk-file>

INFILE <disk-file> KEY [Section] <value> Tests for the presence of the given key in section in <disk-file>

INFILE <disk-file> VALUE [Section]Key <value> Tests if the given section/key has a value of <value> in <disk-file>

/****** NOT CURRENTLY IMPLEMENTED *****/

INFILE <disk-file> CONTAINS SECTION [Section] <value> Tests if in the [Section] of the <disk-file> a string found who matches <value>

is

INFILE <disk-file> CONTAINS KEY [Section]key <value> Tests if in the [Section]key of the <disk-file> found who matches <value>

a string is

-----*/

ININI SECTION <value> Tests for the presence of the given section

ININI KEY [Section] <value> Tests for the presence of the given key in section

ININI VALUE [Section]Key <value> Tests if the given section/key has a value of <value>

/****** NOT CURRENTLY IMPLEMENTED *****/

ININI CONTAINS SECTION [Section] <value> Tests if in the [Section] of the <disk-file> a string is found who matches <value>

ININI CONTAINS KEY [Section]key <value> Tests if in the [Section]key of the <disk-file> a string is found who matches <value>

-----*/

Conditional execution (Old Style)

You may have multiple IF commands available to execute command lines based on external values.

IFMEMBER <GroupName> <Normal commandline>
Is executed if User is member of the give Netware Group. The User must be logged in the server.

IFNMEMBER <GroupName> <Normal Commandline>
As IFMEMBER, but the commandline is only executed when the user is NOT member of the group

IFENV <%EnvVar%> <Value> <Normal commandline>
When the environment variable %EnvVar% is equal to <Value>, the command is executed

IFNENV <%EnvVar%> <Value> <Normal commandline>
Same as IFENV, but NOT EQUAL

IFEXISTENV <EnvVar> <Normal commandline>

When the environment variable EnvVar exists, the command is executed. Please note, that the name of the environment variable is specified without any % signs.

IFNEXISTENV <EnvVar> <Normal commandline>

Same as IFEXISTENV, but NOT EXISTS

Environment Values

In ALL strings, you can insert a sequence %...%, to replace the text with the content of a DOS environment variable.

+`[boot]run=F:\USER\%USER%\WINDOWS\CAL.EXE`

%USER% will be replaced with the value of the USER environment variable.

When It finds a single % in the line the % remains unmodified.

All %% are replaced with a single %.

Note:

You can use F:\USR\%%USER%\TEMP to be transformed into F:\USR\%USER%\TEMP. If you simply use F:\USR\%USER%\TEMP it will try to replace %USER% with the corresponding environment variable.

If the specified environment variable does not exist, a warning message is displayed on the screen, and the %...% is removed from the value.

OS/2

The VIDEO and MONITOR IF-Statements, do not work under OS/2, since this info is not available for OS/2 applications.

EXAMPLES

Please look at the different CMD and ICM files included.

!`[Section]Key=ValueZ`

```
[Section]
Key1=Value1
Key2=Value2
Key3=Value3
```

```
[Section]
Key1=Value1
Key2=Value2
Key3=Value3
Key4=ValueZ
```

>`[Section]Key=Value1`

```
[Section]
Key=Value0
```

```
[Section]
Key=Value0 Value1
```

>`[Section]Key=,Value1`

[Section] [Section]
Key=Value0 Key=Value0,Value1

/[Section]Key=Value1,

[Section] [Section]
Key=Value0 Key=Value1,Value0

<[Section]Key=Value1

[Section] [Section]
Key=Value0,Value1 Key=Value0

IFMEMBER MANAGER ![Groups]Group=LAFW.GRP

Adds GroupXX=LAFW.GRP only if the user is member of the group MANAGER

IFENV %USER% SSCA +[Windows]Supervisor=True

Sets Supervisor=True, when the environment variable USER is equal to SSCA

+ [boot]run=F:\USER%\%USER%\WINDOWS\CAL.EXE

%USER% will be replaced with the value of the USER - Environment variable.

+ [boot]run=F:\USER\%%USER%%\WINDOWS\CAL.EXE

gives run=F:\USER%\%USER%\WINDOWS\CAL.EXE

r[[OldSectionName]=[NewSectionName]

This command renames a section heading. Mtwinn then looks in each line for the Old-name and if found, it replaces it with the new name. You should include the [...] to be sure that only section headings are replaced.

Take care when...

- Modifying **device=** lines in the SYSTEM.INI file. This is the only place where multiple keys exist that have the same name. To handle this special situation you **MUST use the & command instead of the + command**. When you do modifications with the + command, then either all device= lines have the same value, or/and some keys are added multiple times.

- With the r command it is very easy to move a program/library to a new location. Of course you must take care to correct all other references to that location. (In the registration database OLE, in DDE-statements and of course the program icon(s) in the program manager.)

Other Useful Information

- When you maintain the SYSTEM.INI File on a network but still want to use permanent swapfiles, you can do the following:

Copy the SPART.PAR and SYSTEM.INI File to the user-Windows directory. Before you copy the it, remove the read-only attribute from SPART.PAR, (ATTRIB -r SPART.PAR from DOS). This method works well until someone changes the swapfile. For example he uses the windows who is

located on the C: drive and not the one on the network, and then changes the size of the swapfile. When the user now restarts the network-windows, then he gets the info that the swapfile may be damaged. This occurs because the info in spart.par does not match the size/location of 386spart.par.

To avoid this blue-screen to be displayed you can do the following:

1. Create a permanent swapfile
2. Copy the SYSTEM.INI to the machine.directory
3. When a user starts WIndows on the network, then copy the system.ini of this machine to the users directory
4. Delete the SPART.PAR in the user directory (It may be read-only !)
5. Delete the 386SPART.PAR on the C: or D: drive (This file is hidden and system)
6. When you now start Windows, then a new swapfile of the specified size (in system.ini) is created on the given drive.

Known Problems

- SETENV sets only the current environment. I still search libraries..... in C or C++

License Agreement

MTWIN is emailware, which means if you find this program useful, you should send me a email.
(Or a nice postcard if you don't have access to a email system)

If you send me a message, then I will inform you of future versions of this utility. If you have some questions, then please tell me which version of MtwIn you use.

My email address:

```
Compuserve      : 100034,3536 Andre Schild
Internet        : neatech@dial.eunet.ch
AT&T Mail       : mhs!csmail!100034.3536
X400            : /c=US/ad=compuserve/pd=csmail/d.id=100034.3536
MCI             : TO: Andre Schild
                 EMS: COMPUSERVE/ MCI ID:281-6320
                 MBX: 100034,3536
MHS             : MAIL@CSERVE {100034,3536}
CC:Mail         : Andre Schild AT NEATECH (Via Compuserve)
```

Postal address: Andre Schild
Pfeidstrasse 8
CH-2555 Bruegg b. Biel
Switzerland