

Defining an Internal Macro

An internal macro has two different formats depend on where it should be executed from: (1) To be executed from the command prompt. (2) To be executed inside of NWSM. The command prompt macro is in free text format and you can give your file any name you like and it can be in any directory you like. On the other hand, the macros to be executed inside of NWSM are subject to certain rules.

Command Prompt Macro

The following is an example of a command prompt macro. It will print a customer numeric edit list for customer numbers 100 – 200 to the screen and then exit NWSM.

```
0101 05
"000100"
"000200"
{enter}
{enter}
%S
%O
{close}
```

Normally, we will name this macro "CUSNLIST.MAC" and place in the MACRO directory which is either:

```
\NSI\MACRO   or
\ELLIOTT\MACRO
```

To run this macro, you will use the following command prompt (or short cut):

```
NW32 02 /F:CUSNLIST.MAC
```

02 stands for the Accounts Receivable module. You must specify the module for which the command prompt macro shall run. If you wish to run a menu item on the main menu that doesn't belong to any module, then you should use "00". For example, the following command will startup the defer processing:

```
NW32 00 0406 %P
```

0406 stands for drop down menu 4 (Util-Setup) and menu item 6 (Defer Processing). Depending on whether manufacturing modules are turned on or not, you may need to use 0306 in the above scenario if you only have accounting and distribution modules turned on. %P stands for Alt-P which is the keystroke for the "Process" button. You may want to backup your entire system without leaving any file opened by deferred processing. You can let Windows scheduler startup defer processing at certain hours after the back up is done. In this example, we execute an internal macro script directly on command line. The command line is limited to 256 characters, therefore, you will have to use the command prompt macro file if your script is longer than the command prompt limitation.

By default, the command prompt macros use MAC as the extension. If you specify a command prompt macro without the extension, the system will assume MAC is the extension. You can also use a macro file from any directory. When you do not specify the directory, then it assumes the macro file is in the MACRO directory.

Syntax for Command Prompt Macro

The command prompt macro must begin with the menu path (to exclude the module portion) first. In the previous example, 0101 stands for customer file maintenance. 05 stands for customer numeric edit list. Once the cursor is in the actual application, you will use the following scripts:

```
“000100”  
“000200”  
{enter}  
{enter}
```

Which corresponding to the numeric edit list:

```
Starting Customer No    000100  
Ending Customer No     000200  
Print Comment?        N (Use enter to accept default)  
  
Any Change?          N (Use enter to accept default)
```

Anything in a double quote stands for the data to be entered into a field and after that, an enter key will be issued automatically. At this moment, the print option window will popup. The following line will cause the printing go to screen:

```
%S          - Alt-S stand for print to screen  
%O          - Alt-O for OK button
```

After the print is send to screen. The {close} will close NWSM and exit.

You may combine this text on one line, as long as they are separated by a space and the maximum length per line is 255 characters.

Special Keys for Internal Macros

The Internal Macro supports the following special keystrokes:

```
{F1} = F1 Key  
{F2} = F2 Key  
{F3} = F3 Key  
{F4} = F4 Key  
{F5} = F5 Key  
{F6} = F6 Key  
{F7} = F7 Key  
{F8} = F8 Key  
{F9} = F9 Key  
{F10} = F10 Key (End)  
{ENTER} = Enter Key  
~ = Enter Key  
{ESC} = Escape Key  
{ESCAPE} = Escape Key  
{UP} = Up Arrow Key  
{DOWN} = Down Arrow Key  
{PGUP} = Page Up Key  
{PGDN} = Page Down Key  
{CLOSE} = Close NWSM and Exit
```

These keystrokes are similar to the external macro, however, a limited subset. These keystrokes are supported by all NWSM screens except GUI windows. The GUI windows are note windows, tickler, help, macro...etc. There are, however, two GUI windows that support certain macro keystrokes – the Print Option Window and the Defer Processing Menu. More GUI windows may be added to the support list in the future.

Print Option Window supports the following macro keys:

- %P = Alt-P (Print to printer)
- %S = Alt-S (Print to screen)
- %D = Alt-D (Print to disk)
- %O = Alt-O (OK button)
- %C = Alt-C (Cancel button)
- {ENTER} = Enter (OK button)

Defer Processing Menu supports the following macro keys:

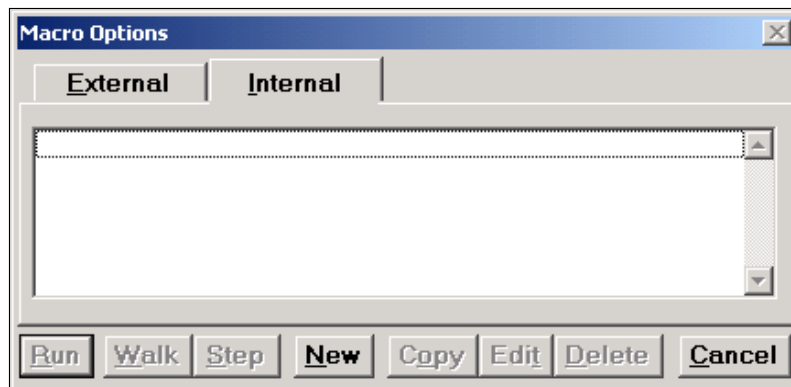
- %P = Alt-P (Process button)
- %C = Alt-C (Cancel button in processing screen)
- %E = Alt-E (Exit button)

Defining an Internal Macro inside an Application

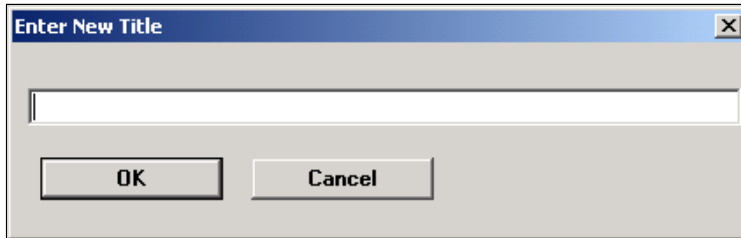
You can create an Internal Macro from any screen where the Macro button is active. Unlike the External Macros which apply to the whole screen (help-id) and you can execute a external macro anywhere in that screen, Internal Macros are cursor sensitive. An internal macro that will execute successfully if the cursor is at phone number field will not work correctly if the current cursor is at fax number. For that reason, when an internal macro is saved, the cursor position for execution is also saved in the macro file heading area. When you try to execute an internal macro, only the internal macro that matches the cursor position will be made available to you.

The following is an example of adding an internal macro in the customer file maintenance screen. The macro will be executed from the “Field Number to Change” position:

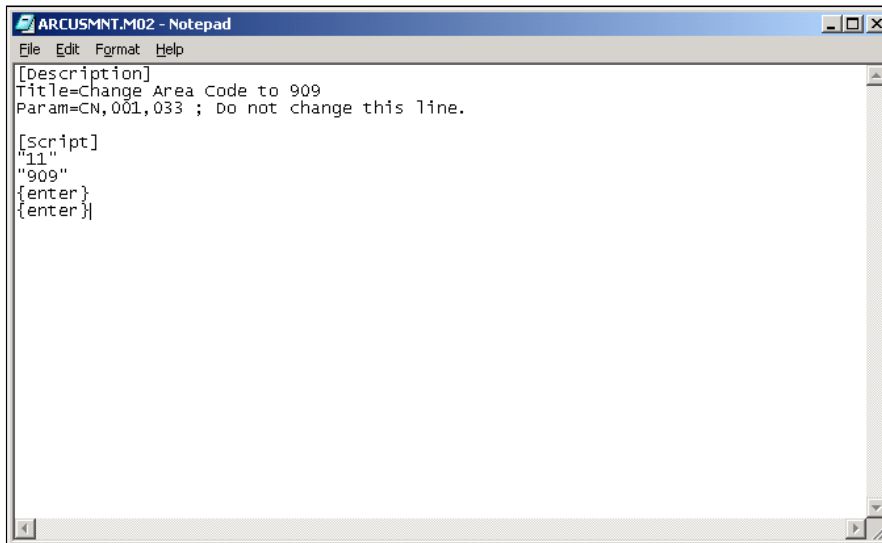
First will go to customer file maintenance and bring up a customer record in change mode, then have the cursor stay at “Field Number to Change”, we then click the Macro button (Ctl-A) and click on the Internal tab. You should see the following screen:



At this moment, you can click on the “New” button and the system will prompt you for the name of this macro:



We can enter, for example, “Change Area Code for 909” and confirm it by clicking OK. After that, you will see the script for the macro display in your favor editor (by default – notepad) like the following screen. Currently, there’s only the heading area and without any script. What you need to do is to enter the following script at the end of this macro:



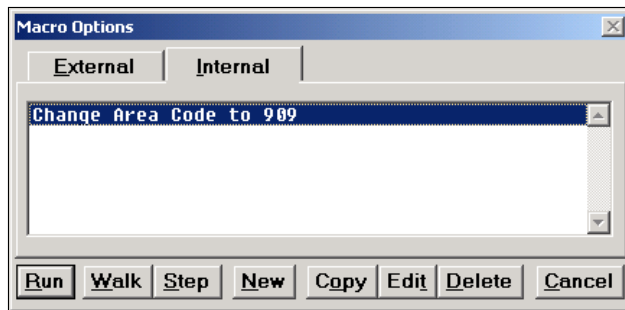
“11” means field 11 (Phone Number).

“909” means change the phone numbers first three digit to 909. You will need to make sure that you are not in the insert mode when you execute this macro.

The first {enter} is to be execute at “Field Number to Change”.

The second {enter} is to respond to the question “Do you wish to display next screen?” which defaults to “N”.

In short, after executing this script, the current customer record’s phone number area code will be changed to 909 and the screen will be cleared. At this moment, save the script in notepad and exit. If you wish to execute this macro, simply go to customer file maintenance and bring up a customer. Let the cursor stay at “Field Number to Change” then click on the macro button (Ctl-a) and click on the internal tab. You should see the following:



Click on the button “Run” to run the macro. If you wish to change it, simply click on the edit button. The internal macro will be saved in the MACRO directory with the naming convention of:

ARCUSMNT.M99

Where ARCUSMNT is the help-id of current screen and 99 is from 01 – 99 sequentially assigned per help-id. For this reason, you can have up to 99 internal macros per help-id.

The syntax for an internal macro to be executed from inside the application is identical to the command prompt macro. There are few differences:

- Command prompt macros require a startup menu path.
- Command prompt macros do not have heading area.

A macro executed in one help-id can change itself to another menu item by supplying the menu path. To exit a menu you will have to supply the menu number for exit in your script, {esc} doesn't work.