Input Message File Format

The input message file is a standard ASCII file that contains three types of lines:

```
  Comment lines
  Component identifier line
  Component message lines
```

Comment Lines

Comment lines are allowed anywhere in the input message file, except between the component identifier and the first message. Comment lines must begin with a semicolon (;) in the first column.

In the Input Message File Example, the comment lines are

```
; This is a sample of an input
; message file for component DOS
; starting with three comment lines.
Component Identifier Line
```

The component-identifier line contains a three-character name identifier that precedes all MKMSGF message numbers.

In the example, the component identifier is DOS. Component-Message Lines

Each component-message line consists of a message header and an ASCII text message.

The message header is comprised of the following parts:

```
  A three-character component identifier
  A four-digit message number
  A single character specifying message type (E, H, I, P, W, ?)
  A colon (:)
  Followed by a blank space.
```

The following message types are used:

```
Type    Meaning
E   Error
H   Help
I   Information
P   Prompt
W   Warning
?   no message assigned to this number
```

Message numbers can start at any number, but messages must be numbered sequentially. If you do not use a message number, you must insert an empty entry in its place in the text file. An empty entry consists of the message number, with ? as the message type, and no text.

The character % has a special meaning when used within the text of a message:

%0 is placed at the end of a prompt (type P) to prevent DosGetMessage from executing a carriage return and line feed. This allows the user to be prompted for input on the same line as the message text.

Mike Note: The %0 can be used with any message type!!!! So here is how this works: The message file is scanned, and each is saved. However, if you place a %0 as the last character the <LF> <CR> is dropped and it does not matter the type of message (E, H, I, P, W, or ?). This is how the IBM MKMSGF worked and my clone works the same way.

%1 - %9 are used to identify variable string insertion within the text of a message. These variables correspond to the Itable and IvCount parameters in the DosGetMessage call.

Component-Message Example

For example, DOS0100E: is DOS error message 100. For additional examples, see the Input Message File Example.

Following is an example of an input message file:

```
; This is a sample of an input
; message file for component MAB
; starting with three comment lines.
MAB
MAB0100E: File not found
MAB0101?:
MAB0102H: Usage: del [drive:][path] filename
MAB0103?:
MAB0104I: %1 files copied
MAB0105W: Warning! All data will be destroyed!
MAB0106?:
MAB0107?:
MAB0108P: Do you wish to apply these patches (Y or N)? %0
MAB0109E: Divide overflow
```

From:
https://ftp.osfree.org/doku/ - **osFree wiki**

Permanent link:
https://ftp.osfree.org/doku/doku.php?id=en:docs:tk:formats:msgsrc&rev=1705679462

Last update: **2024/01/19 15:51**