Disgruntled Postal Worker

```
* Version 1.1Ω ©1995 by Jonathan Guyer <j-guyer@nwu.edu>

* A mail sorter for Eudora 1.5.1 (and later?)
* (for those of us who are too cheap to buy the commercial
version).
* It sorts all mail in the "In" and/or "Out" box, according to an

* instruction file.

* In addition to a basic AppleScript installation, you should
only need
* to ensure that Apple's Scripting Additions has been installed
* (specifically, the "Read/Write Commands" OSAX).
* DPW does not rely on a Scriptable Finder.
* The EudoraNotify.osax will help readability, but is not needed
for the script
* to function.

* If you make changes to the instruction file, you must quit and
restart DPW.
*
* The instruction file must be called "Sorting_Instructions"
* and be in the same folder as DPW.

* Version History:
*   (WARNING!!: this script is so alpha, it's omega.
*               I guarantee/warant/pledge/promise/assure
Nothing!!
*               That said, enjoy!)
*
* 1.1Ω   3/31/95
*     Enabled sorting of "Out" box
*     Enabled sorting of entire mailbox on startup
*     Renamed flags to hopefully make a little more sense
*     Added a AppleEvent TimeOut delay variable to help ensure
that the first
*       mail on startup gets sorted
*
* 1.0Ω3  2/27/95
*     Renamed input file to "Sorting_Instructions"
*     Added #sortAllNewMessages# flag
*     Stopped sorting entire "In" box
*
* 1.0Ω2  2/26/95
*     Moved instructions file from  to folder with DPW
```

```
*
* 1.0Ω   2/24/95
*     Humble beginnings

* This script was inspired by example scripts by Chuck Shotton
* (author of MacHTTP) onn the MacHTTP LISTSERVER and from the
Qualcomm example
* scripts file
<ftp://www.qualcomm.com/mac/quest/eudora/mac/scripts/>

* If Eudora is currently a startup application, it is recommended
that this
* script (or an alias to it) be put in the Startup Items folder
instead.
* This ensures that Disgruntled Postal Worker is stalking the
Post Office before
* any mail comes in (you don't want to get it upset, do you?).
* Eudora will be launched by DPW.

* I've introduced the following flags:

#sortUnreadMessagesOnly#

* If this flag is present, only unread mail will be sorted, all
new mail gets
* sorted, otherwise all new mail gets sorted.
* The second case (without the flag) is useful if you
* receive a lot of mail and you start reading it while DPW is
still sorting;
* this way it will get sorted anyway
* (although, the message you are reading may get snatched away).

* #sortWholeBox#

* This flag saves having to run a whole separate script to get
your life in
* order for the first time.
* If this flag is present, the entire mail box ("In" and/or
"Out", depending on
* the next two flags) will get sorted on startup. If
#sortUnreadMessagesOnly# is
* set, then only the unread messages will get sorted. I recommend
setting this
* flag the first time you run DPW and then commenting it out. You
shouldn't ever
* need to use it again, but it's nice to know that it's there
just in case
```

```
* something causes your mail to not get sorted.

#sortInBox#
#sortOutBox#

* These two flags should be pretty self-explanatory.
* See the following to determine how each box gets sorted.


* The instructions file consists of a series of tab delimited
instructions:
* <IN FIELD> <TAB> <OUT FIELD> <TAB> <STRING> <TAB> <PATH>
*
* <IN FIELD>
 is an RFC 822 header field for the "In" box. If Eudora can't
*             understand it, the instruction is ignored.
* <OUT FIELD> is an RFC 822 header field for the "Out" box. If
Eudora can't
*             understand it, the instruction is ignored.
* <STRING>
   is a search string. If <FIELD> of an incoming message contains

*             <STRING> then the message is moved to the mailbox
indicated by
*             <PATH>.
* <PATH>
     consists of [<MAIL FOLDER> <TAB>]<MAILBOX>. <MAIL FOLDER>'s
*



          can be nested arbitrarily deep. If <PATH> doesn't
resolve to
*             something Eudora knows, the transfer instruction is
ignored.

* Any line starting with an '*' is considered a comment and is
ignored.
* NOTE: You can reduce the load time of this file significantly
by removing
*       all of these comment lines.
*       Make a copy first!!!
*       On my PMac 7100/66 it's the difference between about 17
secs with
*       comments and 2 seconds without.
*       The time to process sorts will not be affected.

* If more than a few sorts are defined, this can be
```

```
excruciatingly slow.
* Put the most heavily used ones (like LISTSERVERS) near the top.

* Does anyone have any suggestions?

* I don't know why, but I can't get Eudora to activate properly:
*
* IF  "Settings:Checking Mail:Check for mail every ___ minutes."
is not blank
*      (you have selected automatic mail checking)
* AND "Settings:Checking Mail:Save password" is not checked,
* THEN Eudora sits in the background waiting to be activated and
have a
*      password entered.
* IF  "Settings:Getting Attention:Use an alert." is not checked,
*      it's quite likely that Eudora will time-out on its
password request
*      before you notice it.
* This isn't a disaster; just not the way I'd like it to work.

* To minimize the problems assosciated with the above, I've added
a timeout delay,
* which defaults to 300 seconds, but which can be set by:

* #startUpDelay#
300

* If the user enters nothing, Eudora's password dialog times out
in about
* 2 minutes. The default setting of 5 minutes allows at least 3
minutes to
* download mail after the user enters a password. If the mail
download takes
* longer, DPW will report a timeout error, but it will sort the
mail anyway.
* If you download lots of messages over a modem line, you might
wish to extend
* this delay, just to avoid having to aknowledge DPW's error
message.
* I'm not aware that there are any particular repurcussions to
setting it
* longer.
* Anybody??

* This flag is only relevant on startup of DPW & Eudora and only
if automatic
* mail checking is selected.
* Without this delay, the default AppleScript delay of 1 minute
caused DPW to
* report a timeout error before Eudora got tired and put its
password dialog
* away. If the user entered a password in the next minute, the
mail sometimes
* did not get sorted (a major pain, I know).




* If a message satisfies more than one of the following
instructions, it will be
* sorted according to the first one. Likewise, if a message in
the "Out" box is
* sent to multiple recipients, it will be sorted according to the
first match. I
* may eventually figure out how to not sort multiple-recipient
outgoing mail.
* Any thoughts?

* To sort mail from LISTSERVERS, use the sender field.
* The mail author (From:...) varies, but sender is the
LISTSERVER.
* Will this always work?

* So, the first instruction below means that messages in the "In"
box that have
* 'owner-machttp-talk@academ.com' in the 'Sender:' field will got
sorted to
* 'Lists:MacHTTP:MacHTTP List'. Likewise, messages in the "Out"
box that have
* 'owner-machttp-talk@academ.com' in the 'To:' field will get
sorted to the
* same mailbox.

sender
to
owner-machttp-talk@academ.com
Lists
MacHTTP
MacHTTP list
sender
to
Macintosh Scripting Systems
Lists
```

```
MacScript
sender
to
nih-image@soils.umn.edu
Lists
NIH Image

* It may arise that the <STRING> for the "In" and "Out" boxes
won't be the same.
* Maybe I have a nickname defined such that mail I address to
'MacGod' gets sent
* to John Norstad, but mail I get from him (I actually do get
mail from John,
* occasionally, and it's cause for much rejoicing by the
peasants) is from
* 'John Norstad'. In this event, I'd need to define two separate
* instructions:


to
MacGod
I'm not worthy
Norstad

* and

from

John Norstad
I'm not worthy
Norstad

* This way, mail both to and from Norstad would get sorted to
* "I'm not worthy:Norstad" (as opposed to "I'm not
worthy:Locklear" or
* "I'm not worthy:Elvis" (I don't get nearly so much mail from
them))

* Note the null field at the beginning of the first instruction
and
* in the second position of the second instruction.
* Instructions with null fields will be ignored by the sorter,
i.e., my
* "In" box will not get searched for messages involving "MacGod".
* In general, it's preferable not to have these double
instructions,
* since each instruction takes some time, whether it's
```

implemented or not.

* Be very careful with spaces and tabs. DPW won't puke on spaces in your paths,
* but it's almost impossible to tell spaces and tabs apart in SimpleText.
* Tabs are valid field delimiters; spaces are not.
* Spaces are valid characters in any of the fields; tabs are not.