Dead or Alive document

COLLABORATORS			
	TITLE : Dead or Alive document		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		February 12, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Dead or Alive document

1.1   Dead or Alive document

```
                                                                        _____  ↩

    _ _____  __ _ _ __ _____            :         |
    \\__     \/ \ensuremath{\lnot}\_           _____ ___      /__ _ __ ___|_____ ↩
       |
    /    __ ___/_____/_  _/ ____/  /   ____/_____  \ensuremath{\lnot ↩
        }\_   |
  __ /   / \__    /_____   \/   /   /     /    /____/___|____
  | /   /   /   /   /   /   /   /   /   /    /    /    /       :
  ¦ /_____/   /   /   /____/_   /   /_____   /   /   /   / ·MYSTiC· ¦
  :____ /_____/_____/___   /___ /_____/   /_____    /_____|
         /    / kRm/_____/         /_____/_   /_____/       |
         /_____/                       /    |        ¦
                                       ·       |_____:
```

-·> DEAD OR ALIVE v1.2 [RELEASE 1] (c)1996 Mamba/Mystic & Zeus/Mystic! <·-

If you are updating from earlier version, read the file "Changes.DOC"!

            Contents
            What is included in this archive

            Features
            What can you do using DOA

            Installation
            How to install

            Configuration
            How to configure

            History
            Revision history

            TODO
            Things to do

```
                    Greetings
                    Thanks & greetings

                    Contact
                    Contacting me
```

1.2   Archive contents

This archive contains following files:

```
    Configs/DeadOrAlive.CFG        Configfile
    Docs/DeadOrAlive.guide         Documentation(this file)
    Doors/DeadOrAlive              The program itself
    Misc/DeadOrAlive.Msg           Message file
    Misc/DeadOrAlive.Msg2          Another message file
    Misc/DeadOrAlive.Msg3          And a message file, again. :)
    Screenshots/Preview?.iff       Screenshots taken from Zeus's terminal
                                   Nice font, isn't it? :)
    Utils/DeadConfig               GUI Configuration Editor
                                   (done fully by Mamba, no nice outlook :)
```

 Sorry, the executable is quite big (C code), but the source is also
 abt 71k long.

1.3   Thanks & greetings

I want to thank Zeus & Loop for nice layout design. Greetings to all my
friends. :)

1.4   Revision History

```
                1.0 - First public release.
```

1.1 - If user's online time ended or something happened to line, jammed.
      Fixed.
    - Some bugs with layout fixed.
    - It is pure (compiled using DICE's option -r).
    - Multinode support. (Option, if set ON, datafile and outputfile
      is read/written every time there is need for it.) Slows a bit.
      Seems to work well, but who knows.. :)
    - Support for long comment to message.

1.2 - GUI configuration editor. Descriptions of the gadgets are in chapter

                Configuration
                . You must replace old configfile with new one
      because I changed order of the configfile to more logical (filenames
      first etc). Some lines removed and/or put to same (eg. there were
      "filedir" and "listclean mode" entries, now there is only "listclean

```
command"). Namemode is now taken from Conferences.DAT. After
you have updated configuration file with included one, use
DeadConfig to edit it. You must update also messages because
I changed their order too.
```

```
- Online configuration. You can change settings online. Nice outlook
  is done by Loop. Use door parameter CONFIG. Use up/down cursors to
  move between entries, enter to select an entry for editing and Q to
  quit & save or A to quit without saving.
```

```
- Again, layout had little bugs. Fixed.
  If handle/something was longer than space allocated for it, it
  was still displayed in full. Change %-xxs to %-xx.yys in messages
  where yy is the maximum lenght of string. (Only if you have edited
  them, otherwise replace old ones by new included ones.)
```

```
- If request (stuff) had %-mark, messed a bit. Fixed. Thanx to Mad Jack
  for reporting that.
```

1.5 Features

```
o Keeps a list of requests (in datafile) and writes them to outputfile.
```

```
o Configuration is done via external GUI config editor or on-line
  config editor. (Configfile is still pure ASCII, manual editing is also
  possible :)
```

```
o User can select action using cursorkeys or by number or by first letter
  of action.
```

```
o Allows users to leave a message when requesting something, so all users
  can see the request "automatically".
```

```
o Users can upload requests via DOA and after upload leave a message
  to requester. "I uploaded your request <stuff>." Then requester knows
  to download that request.
```

```
o The layout of these messages is editable by sysop. Of course sysop
  can turn these messages off if he doesn't want them.
```

```
o Users can also download requests via DOA. If requestconf has
  CONF_FREELEECH on, requests downloaded from DOA are free download.
```

```
o Inbuild "faker". When user has downloaded a request and his/her
  accesslevel is high enough (change the needed acclevel from config),
  he/she is asked if he/she wants to reward uploader with extra credits.
  The maximum factor for extra credits is also definable by sysop.
```

```
o Allows users to delete own requests. If user's accesslevel is high
  enough (this one in config too..) he/she can delete all requests.
  If deleted request was uploaded, DOA also deletes the file and runs
  ListClean, so it will be removed from filelist, too. Saves some
  work from sysop! :)
```

o Creates a logfile is sysop wants it.

o Works with multinode systems too. Other DOA will notify changes in
  "realtime".

o Can be made resident.

1.6 Things to do

– Editing requests.. Is there need for it? It would take one line from the
  "main menu" and then I must ask Zeus for new layout etc.. :)

– Perhaps help page to online config...?

– If you have any comments/suggestions/whatever, tell me and I'll try
  to fix/add/do it.

1.7 Contact me

Leave a message to Mamba at one of the following boards:

  Cloudkill, Zillion Hours, Hard Disk Cafe, Bermuda Triangle, Fear
  Factory, The Chameleon .. or any other board I have access to.

 Finns by phone +358–54–443181 / Sampo. (Fastest way..)

 E-mail: mamba@tup.nullnet.fi.

1.8 Installation

            1) Copy files to their directories in DayDream. Just go to ←
               directory
  where you unpacked the archive and type:

   copy #? DayDream: all

  Or, I have stuff like the outlook of the messages etc in DayDream:Misc,
  if you don't want to create it, you must edit the configuration.
  Because there at least are 4 files, I decided to keep them in
  Misc-directory because I didn't know where to place them. Ofcoz,
  you can change them in configuration. The configfile is searched from
  path specified at ENV:DreamUtils, as Hydra recommended to do.

2) Add something like this to ExternalCommands.DAT:

DOOR_COMMAND.. REQ
DOOR_TYPE..... 1
DOOR_SECURITY. 5
DOOR_EXECUTE.. Doors:DeadOrAlive %N
DOOR_CONFS1... XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```
DOOR_CONFS2... XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
Then you should
                configurate
                 and test it.
```

1.9 Configuration

There is now an external configuration editor, DeadConfig. It is ↩
                much more
confortable to use it than edit ascii-files. :) Perhaps I should explain
the gadgets better, because I had to keep the strings in program itself
short.

NOTE: You perhaps don't have the Data & Outputfiles when you are running
      DOA first time, so filerequester can be unnecessary.

```
Datafile  : Requests are stored there in internal format.
Message 1 : Message (Entering request, no long comment)
Message 2 : Message (Entering request, long comment)
Message 3 : Message (Uploading request)
Outputfile: ASCII/ANSI request file (Put it to JoinConference.DAT :)
Author    : Author for message when Entering request. Can be DeadOrAlive!
            or something like it. Set it to "Requester" if you want that
            the person who is making is put to author.
Receiver  : Receiver of the message. (When entering request)
1. Subject: Subject for message (Entering request)
2. Subject: Subject for message (Uploading request)
LC Command: ListClean command. This command is executed when user has
            deleted uploaded request(s). The files will be removed from
            filelist, it saves a little work from sysop.
```

 The "Get" gadgets open a filerequester and you can select the file using
 it. It is perhaps not usefully when running DOA first time, because you
 don't have output & data files then.

        Switches:

```
Allow msg 1: Allow users to send a message when entering request.
Allow msg 2: Allow users to send a message when uploading request.
Multinode  : Multinode system. If TRUE, outputfile, datafile etc. are
             written more often, so DOA in other node can notice changes
             in realtime.
Logwrites  : Write "important" events to DayDream:Logfiles/DeadOrAlive.Log.
```

        Others:

```
Message Cnf: Conference for the messages.
Msg Base   : Base for the messages.
Del any req: Accesslevel needed to delete any or all requests.
Reward user: Accesslevel needed to give extra credits for user.
Max factor : Maximum factor for extra credits.
Config Acc : Accesslevel needed to onlineconfig.
```

Dead or Alive has inbuild "faker". It means that users can reward other
users easily from this door. Sysop can define what is the top limit for
credits (Max factor). When user has downloaded a request and his/her
accesslevel is same or higher than accesslevel defined "Reward user" at
config, he/she is asked if he/she wants to give extra credits to uploader.

Requirement for this is that both, uploader & downloader have used DOA
for uploading and downloading request.

Also a little message is sent to rewarded user ("You got x*credits for
uploading <stuff> to <requester>").

Messages can contain some of the following codes:

        Message 1:
        ~~~~~~~~~~

    %w (what)   What is requested
    %r (requester) Who requests
    %d (date)   Date of request.

        Message 2:
        ~~~~~~~~~~
    Same as message 1, but also space for additional comment.

    %a1 Line 1 of addit. comment.
    %a2 Line 2 of addit. comment.

   1 or 2 after 'a' are changed to space (' '), not removed.

        Message 3:
        ~~~~~~~~~~
    %u (uploader)   Who uploaded request
    %w (what uploaded) What uploaded

See some
                notes.

1.10 Some notes

In messages, you can use also following forms: (for example)

    %-26.26w   Means that requested item (w=what) takes always 26 chars
               space. This makes for example Msg1 possible.. If you need
               less than 10 chars, use form %-08.08d.. or something like it.

If you have handles in use and you have some requests there and you change
the mode to realnames, older requests still have handles in the list. It's
possible to do it so that there are stored both, and if you have handles
on, handles will displayed and if you have realnames then realnames will
displayed. But in my opinion it isn't worth of doing, because in other
systems handles are used everywhere and in others nowhere. And it would
take more memory and disk space... :)

You want to know how downloading and uploading are done? When user selects
"4 Upload Request", he/she will first asked the number of request he/she
wants to upload. When he/she has answered it and given request exists,
Dead or Alive firsts checks if user is in requestconf(number specified in
config). If yes, "U" is executed as an internal DayDream command. Otherwise
Dead or Alive calls JoinConference() with flags JOINCONFF_QUICK and
JOINCONFF_SKIPTXT (no conference data is loaded and no JoinConference.TXT/
GFX are shown). Then it executes "U" as an internal command.

When uploading has ended, Dead or Alive looks the last entry of
DayDream:Logfiles/UploadLog.DAT. If conference, user slot number and node
number match with values in uploadlog.dat, request will marked uploaded
and filename is got from uploadlog.dat. If the values (slot,node,conf)
aren't same, DOA asks the filename from user. User can tell to DOA that
transfer failed by pressing only <enter> when asking filename.

That can cause some problems if you have many nodes and someone uploads
stuff same time in other node.. :) But it shouldn't cause problems because
Uploadlog.DAT is opened immediately after upload has ended and DayDream
has wrote it.

When upload has ended correctly, user is asked if
he/she wants to send a message to requester style "I, mr.x uploaded your
request <something>." (user won't be asked about it if you have disabled
it in config). You can change the outlook of that message, ofcoz. :)
Then door returns to "main menu".

When downloading, Dead Or Alive first checks what current user's requests
are uploaded and shows a list of them. User can select what number to
download, and if the given number is uploaded, the file is flagged for
downloading. If the requestconf has CONF_FREELEECH on, the file will be
free download. Then internal command "D" is executed. When downloaded,
user will be asked if he/she wants to delete the request. If he/she answers
Yes, Dead Or Alive deletes the file that user downloaded and executes
ListClean command (defined in config) If there were other deleted files in
that conf, they will marked deleted too. NOTE: ListClean is run when
quitting, so files can be seen in the directory.xxx from other nodes
before Dead or Alive is quitted.