DFB

COLLABORATORS			
	TITLE : DFB		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		August 5, 2022	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

1 DFB **1**

1.1 DFB V1.3 Documentation . . . 1
1.2 Disclaimer . . . 2
1.3 Utilities . . . 2
1.4 Disclaimer . . . 2
1.5 History . . . 2
1.6 Copyright . . . 7
1.7 Requirements . . . 7
1.8 Introduction . . . 7
1.9 Installation . . . 8
1.10 How To Use . . . 9
1.11 What's new for V1.2/1.3 . . . 12
1.12 Developers . . . 12
1.13 DFBPrefs . . . 13
1.14 DFBSection . . . 15
1.15 DFBSection - File Manager . . . 17
1.16 DFBSection - File Ratios . . . 18
1.17 ReadDFB . . . 19
1.18 ReadDFB - How to install . . . 19
1.19 ReadDFB - User functions . . . 22
1.20 ReadDFB - Sysop functions . . . 23
1.21 ReadDFB - Customization . . . 25
1.22 ReadDFB - Headers . . . 25
1.23 DFBArchivers . . . 26
1.24 DFBArchivers - File backups . . . 27
1.25 DFBLogin . . . 28
1.26 DFBArchivers - File_Id extractors . . . 29
1.27 DFBArchivers - Viewing files . . . 29
1.28 DFBArchivers - What to do when a file is uploaded . . . 30
1.29 DFBUser . . . 30

1.30 DFBUser - When to user DFB's editor, when to user MAX's 31
1.31 DFBUser - What everything does 31
1.32 DFBUser - User Manager 32
1.33 Feature List 33
1.34 Credits & Greetings 34
1.35 Future of DFB 35
1.36 How to Register 35
1.37 Congratulations 37
1.38 Registration Form 37

Chapter 1

DFB

1.1 DFB V1.3 Documentation

Welcome to the Dynamic File Base user's manual

Select on any topic to continue...

Initial... Programmes...

Disclaimer

ReadDFB

Copyright

DFBPrefs

Requirements

DFBSection

Introduction

DFBUser

Installation

DFBArchivers

How to use/Hints/HELP

DFBLogin

What's new for V1.2/1.3

Utilities
Final...

Feature list

Future of DFB

Credits & Greetings

Developers

Registration

History

1.2 Disclaimer

This programme and its documentation is provided 'as is', the author cannot be held liable or responsible for any damage which might be caused directly or indirectly by the use or misuse of this programme, any damage caused is the sole responsibility of the user. So nyaah.

1.3 Utilities

Files in the utilities drawer:-

File	Brief Description	Author
DFBFlg	A file list generator for DFB.	Me
DFBCheck	Keeps track of DFB/HD files.	Me
DFB_UserStats	Displays users' statistics using DFB.	Prowler/Cydonia
DFB_TagShow	Lists tagged files and lets user delete them.	Greg Fitch
DFB_TagKill	Kills users' tagged file list.	Greg Fitch

1.4 Disclaimer

This programme and its documentation is provided 'as is', the author cannot be held liable or responsible for any damage which might be caused directly or indirectly by the use or misuse of this programme, any damage caused is the sole responsibility of the user. So nyaah.

1.5 History

Version	Date	Comment
V1.0	14/02/95	Initial release.
V1.0a	19/02/95	Thanks to Greg Fitch, for doing some impromptu beta-testing of DFB on a 68000, who located many bugs!

		Bug fixes in ReadDFB now means it can also run on a 68000 based machine. A problem with one of DFBUser's requester has now been fixed. In DFB.library TimeToAscii & DateToAscii have been fixed, also an oversight in GetUser has been fixed, a routine to Add/Modify user has been enhanced. In DFBSection, clone section now works on a 68000 also a small bug which could save the wrong readings of the cycle gadgets has been fixed. DFBPrefs, when creating the file base and file_id.dizs are off, no longer checks whether the files exist or not - speeding up installation time enormously.
v1.0b	05/03/95	In DFBPrefs ratios 2 & 3 are now around the right way for enhanced members. DFBUser should now be more stable. Now there is only one registered file to get: DFB.library DFBArchivers now doesn't crash if you try to restore or backup files when you haven't set the backup file name. When checking a user's information from ReadDFB the first conference ratio's stats are shown as opposed to the sum of all the conference ratios (which was generally a useless result and didn't add up anyway). When changing file listing direction in ReadDFB it missed a file - this has now been fixed. ReadDFB now excepts the numbers 8,2,4,6 as up, down left and right for those that don't have arrow keys.
v1.0c	02/04/95	Edited requesters, now should hopefully work with ARQ. Fixed - listing files after date; also if a user is sleep disconnected after uploading files the files are now added to the file base. (only aliased sections had this problem before.). A bug with reverse listing introduced in v1.0b has been fixed. Now handles MAX's % control codes - adding them to the file name, uploader, alias or file description will not activate the commands. Now displays user's upload/download info before and after downloads and also after uploads. Now displays number of files listed and total files after each file listing. The remote file editor can now zot files and nuke uploader credits. Added 'save' and 'next ->' buttons to the user and section editors.
v1.1	15/09/95	Re-coded the reading routines, ReadDFB is now up to 400%

faster when listing files, also ReadDFB should now be as stable as a stapled down rock.

DFB.library now has expunge code in it for tight memory users (flushlibs from debug menu on Workbench will now remove it).

The 'i' key in ReadDFB now displays more information including how many files listed so far and the total number of files.

The local file editor can now change the file size.

The local upload requester now remembers your last upload directory so you do not have to keep setting it every time you use it. Similarly with the position of the 'go to user' gadget in the user editor.

ReadDFB only asks for upload alias and displays upload information if more than 0 files were transferred on upload.

DFB now does not crash if it loads in a tagged file list (DFBML. file) of 0 bytes.

DFBArchivers now has a 'default' option so DFB can process files with no extensions. (Thus viewing files online with no extension is possible.)

Users can no longer upload to a section which is offline. Hotkeys added to GUI's! Example: <esc> exits windows

The displaying of uploader/alias now works _perfectly_!

DFB now supports headers! You can now add a different text message at the start of every file list and also you can change the colours every file list.

Hopefully the infamous '155' problem is now fixed and even if it hasn't ReadDFB has some 'smart' code to,reduce (read eliminate) the problem.

Created the layout file: longfileid.lay which allows up to 20 line file_id.dizs.

DFB now does not crash if it encounters file names of 0 bytes (this did happen!). (This IS the reason that the release of DFB 1.1 was delayed).

DFB now has code to detect corrupted files in the file list and cleans them up before it displays them.

Wrote the DOS command Zot which can remove files from the file list so you can remove files without entering ReadDFB.

Wrote DFB's File List Generator programme - DFBFlg.

(Now available in the utilities directory)

v1.11 12/01/96 The BenchMaster release... :-)

'ReadDFB last R' now works properly !
Speeded up 'ReadDFB last' so it gets to displaying the new files a lot quicker. No more waiting 10 seconds to be told that there are no new files!

Fixed the standardshort.lay so there is now always a space between date and size of file.

Created a new layout file:- shortnouploader.lay
Modified the 'header' code, should now work properly.

Downloads now do not convert the file names to lower case.

You can now edit the N,L,O,D and + flags when using the ReadDFB N, ReadDFB O etc options.

DFB can now extract file descriptions from file comments (filenotes).

DFB's file description editor has FINALLY been finished! For those that don't rely heavily on file_id.dizs.

Lots of minor fixes including the 'one weird character displayed' bug. Also fixed the rare '28-char bug'.

Created a new layout file:- longdesc.lay

Added DFBPrefs purge CLI option.

Fixed a 1.1 AddFile()/Installation bug.

v1.2 23/05/96 Changed 'marked' to 'tagged'. Now says 'Searching...'

DFB now handles 'missing' libraries properly, including DFB.library and very low memory situations.

Fixed a bug which caused the screen length to be set to zero.

DFB now supports free file sections.

Changed the registration file method for the second time. Registration now comes in the form of a keyfile.

Wrote DFBMakeKeyFile so registered users can use the public release and not need to get anything from me.

DFB is now 1 year old and still going strong

Round 2: Andrew Leppard vs. 199 problem. Now DFB changes the pause flag not the screen length.

		New command in login will now totally remove the 199 / pause flag problem. (DFBLogin) If a user uploaded files without leaving file descriptions, DFB will now get them to enter the file descriptions at login. (DFBLogin) DFB now handles corrupt user bases a little better. Wrote DFBCheck to keep track of your BBS files - a CheckFiles replacement for MAX's. The manual has been spell- and grammar-checked by The BenchMaster.
v1.21	04/07/96	dfb.library now uses MODE_OLDFILE instead of MODE_READWRITE. The latter doesn't seem to work the same for AFS. DFB now works with Ami-FileSafe! Not an official release!
V1.3	13/04/97	The sysop can now edit files through a remotely connected user's account. No matter what the user's access level. Improved multiline synchronization, DFB now has waits around certain commands to prevent two nodes running them at the same time. DFB now deletes file protected file_id.diz's after it has extracted them. Wrote MaxsPatch to allow programmers to take over MAX's serial device, it also fixes the pens to make MAXS use the WB2 3D window look. Totally rewrote the local file editor, now takes advantage of the 'new look' WB 2. Moved the window's 9k of old code into the library to lower memory needed for multiline BBSes & speed up loading of ReadDFB. Another result of this rewrite is that the checkboxes *finally* look right (they used to look weird but always worked). Also the text section search now works. Added a Save button and a new Needs Description checkbox. Now a copy of the preferences is kept in memory, so the preferences aren't loaded in everytime ReadDFB is run. This speeds up loading and decreases memory size by anywhere up to 25k for a large BBS per node. Added click to do 'Nothing' buttons on DFBCheck so people who click a lot won't accidently delete files. Fixed the PurgeFileBase menu option, it never ever worked. Now added a requester to tell the user if he needs to close down any DFB programmes for the purge menu to work.

Also added a 'Click to edit' button which allows users
to editfiles from DFBCheck.

DFBFlg can now generate file lists in Amigaguide and
HTML format.

DFB now has an installer/updater script for the
C= installer programme.

DFB is two years old!

1.6 Copyright

DFB is copyright 1994-7 by Andrew Leppard. DFB may be freely copied as
long as the following conditions are upheld:

1) No 'registered' file of DFB is copied/re-sold/lent/borrowed to other(s).
   Where 'registered' refers to a DFB.library file which has downloading
   enabled, or a DFB key file.

2) No fees are charged in the copying process, non-profiting distribution
   fees exempt.

3) Authorization must be given for this programme to appear on cover discs,
   magazines etc. Fred Fish and Aminet already have my permission to
   carry this programme.

4) None of the files are to be modified or split up.

1.7 Requirements

DFB requires WB2.04 or higher, it requires MAX's BBS running with (initially)
at least one file section. It's recommended that MAX's BBSs running DFB
should have at least 1.5 megs of memory. Also at least an 8 colour Workbench
is recommended for viewing the icons :-)

1.8 Introduction

Liquid Softworks is proud to present V1.3 of this programme...

Firstly, welcome to DFB the dynamic file base system for MAX'S. I must say
now that if you plan to use this programme I *strongly* recommend you to
read _ALL_ of this manual - some features are NOT obvious and may do things
other than you originally thought they would. (Its a case of RTFM again)

So what is DFB? DFB is a complete file base replacement system for MAX'S
BBS. Many doors and utilities have come about offering enhancements for
MAX'S namely: free file downloads, log off after transfers, byte ratios,
file_id.diz support. DFB has all of these features and plenty more. There
is a

```
               Feature List
                of DFB included in this documentation for exactly what
DFB can do. For what DFB can't do see the
               Future
                section.
```

1.9 Installation

```
                Installing DFB is relatively simple to do, just follow these ↩
                   steps:
(Or run the given install script!)

1) DFB must know where to find its files, so before running any DFB
   programme you must have these directories assigned:

   (Add the assign commands to your user-startup, such as:

   assign DFBdata: dh2:maxs/doors/DFB/

   which means DFB will keep its file data file in the directory
   dh2:maxs/doors/DFB/ )

   DFB:       This contains the programmes:
               DFBPrefs
               ,
               DFBUser
               ,
               DFBSection

               DFBArchivers
                &
               ReadDFB
                   DFBdata:   Contains the data file DFB.data (created on ↩
                      installation)

   DFBPrefs:  Contains the data file DFB.prefs          ''
   DFBusers:  Contains the data file DFB.users          ''
   DFBtag:    Contains the users tagged files

2) Then type from CLI:

   copy DFB.library libs:

   Also copy all of these files: DFBuser, DFBsection, DFBarchivers, DFBPrefs
   & ReadDFB to the path you assigned DFB: to.

3) Reset the computer so the assign commands are run.

4) FOR CONVERTING MAX's BBS files over to DFB:

   Now run DFBPrefs and follow its instructions. DFB will tell you that
   it is not installed then it will allow you to edit your preferences.
   To install DFB just exit the programme and confirm that you wish to
   install.
```

Warning! Before you install DFB (step 4) make sure the following are set right:

1) Ratio type: File,Byte or Credit. After installation you cannot change this.
2) Use File_id.dizs: If you want to use them make sure this is set for obvious reasons! You can change this flag after installation if you wish.
3) Go into Archivers, if you are using file_id.dizs's fill out the two left-most columns now.

5) Now edit MAX'S BBS's menu text so it runs ReadDFB instead of referring to the old MAX'S file base. See
ReadDFB installation
for more details.

6) Install
DFBLogin
, to do so: copy the file DFBLogin to BBS:Doors/DFBLogin. Then add to the file BBS:Doors/IntroDoor.text the line

DFBLogin

7) You can add extra files in (i.e., not from MAX's BBS) using standard DFB upload functions. Such as: using the command 'ReadDFB up' which puts DFB into direct upload mode. You can then upload files in using the asl requester.

If you have the flag 'Use File_id.dizs' selected (in DFBPrefs) DFB will then try and use file_id.diz's as file descriptions.

If it can't find a file_id.diz or you do not wish to use file_id.dizs DFB will then check to see if the file has a filenote (ie a 'comment' displayed when you list the file from the CLI), if it does have one DFB will then ask you whether you wish to use that as the file description.

(Note: DFB will only ask this question once per 'batch' upload, ie if you upload 200 files in one go, DFB will only ask you the first time it encounters a file comment - it will then assume that what you answered applies to the remaining 200 files.)

1.10 How To Use

After installation you still have to configure MAX'S BBS to use the DFB files. The file
ReadDFB
is the MAX'S front end for DFB, this programme will list files, upload files, download files etc. Put this in the menu boxes and run it as you would run any other door programme.

ALSO: it is necessary to delete line 182 from your MAX'S config files, this line is a 'press return to continue' and occurs after download. If you do not do this DFB will send files one by one and wait for the user to

press return after each.

It is recommended for your users to have auto-download enabled so that their files are not transferred one by one.

DFBPrefs
is a general preferences front-end. It includes buttons so you can immediately jump to the following four editors:

DFBSection
is the section editor, if you need to add a section, modify a section or delete a section do it with this. After installation DFB ignores the MAX'S sections. You need at least one file section at all times!

DFBUser
is the user editor, if you need to change a user's file ratio, upload signature etc. use this. Do not edit any 'file' information from the MAX'S user editor - this will not be recognized by DFB, DFB only reads this information at installation time.

DFBArchivers
controls the archivers used with DFB. It also controls the following:

1) What command to execute to extract file_id.dizs
2) What command to execute when someone uploads a file (Example: should DFB add in a 'bbs ad'?)
3) What command to use to view files (i.e., when a user wants to view a file what should DFB do?)
4) Automatic back-up of DFB files
Run the File Manager at least once a week - this will keep your files in order.

DFBLogin
makes sure the user's pause flag is on and also checks to see if the user uploaded any files without leaving a file description. If the user did it gets the file descriptions and updates the file base.

Problems

1) A few people have found problems with DFB not finding files (generally in certain sections).

First of all, when DFB uploads a file it uses the file path found in the DFB section editor. When DFB searches for a file it uses MAX'S File paths.

Remember to include trailing '/'s on file paths in both programmes!
Also if you do edit MAX'S file paths you *must* reset your computer before
DFB notices the changes or select flushlibs from the Workbench menu
(this requires you to bootup Workbench with the loadwb -debug option).

2) If you're a registered user and you are wondering why some of the new
features aren't kicking in (Such as the 'default' gadget in DFBarchivers)
it may be due to the fact that you have an 'old' DFB.library. The release
1.1 is considered the minimum. Doing a version command check that your
library is at least version: 1.6. (NOTE: the version command gives a
DIFFERENT reading than the release version, release 1.1 will give version
1.6 on DFB.library. Confused?)

3) You've just installed V1.2 of DFB and are ready to go, you list
files and WEIRD!!! The description has combined with the date??? What's
going on??? Well a minor problem in V1.1 and below was overlooked in
the design stage - but LOOKED ok in most layouts. Basically this has
been fixed and the layouts have been modified to look ok. This means
you may need to use the new layouts included in this release.

How to do:-

Go into DFBPrefs, go to the layout box you wish to change. Change one
letter from upper->lower case or vice versa. This tells DFBPrefs to
'reload' the layout file into the preferences settings.

(The upper->lower case trick 'fools' DFB into thinking it's a different
layout so it goes to load the layout again.)

4) SHOCK HORROR! Here's the scoop shoppers, you are wondering lonely
as a cloud, you spot one of your friendly files encumbered with
a _wait for it_ CORRUPT file_id.diz!!! But it gets worse, DFB.. in a
fit of RAGE exits from ReadDFB (it does this when it gets REALLY confused)
so you figure AHA I'll just delete this file from my DFB.data file!

Then you realize, if I can't tag the file... because it exits, how do I
kill it?? Well new for V1.1 I have written a command called Zot.
(It's in your c directory in the DFB archive)

You can run it like:

zot <file>

and then this command will hunt down the offending file and remove it
from the file list. Simple.

Housekeeping

It is important to 'purge' the file base regularly, this tidies up
the file base and removes any possible corruptions in it. If you find
DFB is acting mysteriously when listing files this may be your best bet.

How often?? Well this depends on your BBS. A two line high-use BBS might
need the file base to be purged daily.

Purging the file base is easy, it is done every time the 'File Manager'
is run. You can purge the file base anytime by selecting 'purge file base'

from the DFBPrefs programme.

1.11 What's new for V1.2/1.3

```
                Major new features for V1.11 & V1.2

* DFB can now extract file comments from files when doing a local upload
* DFB now has 'free file sections' so you can make a file section
  automatically contain free downloads.
* The DFB archive now contains a utilities drawer which contains some of
  the latest DFB
               utilities
               .
* The DFB archive now contains the developer's kit
* New DFB utility -
               DFBLogin
                - gets file descriptions of files which were
  uploaded but no description was left.
* New DFB utility - DFBCheck - which keeps track of your BBS files. It is
  a replacement for the MAX's CheckFiles menu option.

(IF you have just upgraded I suggest you go to
               DFBLogin
                now.)

New features for V1.3

* ReadDFB's local file editor has been totally rewritten. It is now
  in the new look WB2 style. Ive added a Save button, also a Needs Desc
  checkbox. If this is clicked the user will be asked at login to
  enter a file description for that file. This is useful if a user has
  left a description you do not like and you wish him to enter another.
  The string gadget next to the section number gadget is a section jump
  string. Enter part of your section name and it will jump to it.
* Assorted speed increases
* Using MaxsPatch, MAX's screens are now public, so the command:
  amigaguide dfb:dfb.guide PUBSCREEN MAXS.1
  will pop up DFB's doucmentation on your node 1 MAXS screen :-)
  (there are also MAXS.2, MAXS.3 etc.. for further nodes)
* Can now edit files from the DFBCheck programme
* File lists can now be generated in Amigaguide and HTML format
* Installer install script for updating/installing

Not everything has been updated in this release, so some programmes
may say V1.2!
```

1.12 Developers

```
DFB is heavily dependent on its library, DFB.library, this library contains
dozens of functions for accessing the file base and associated features.

Programming and developing for DFB using this library makes life much much
```

easier.

A developer's kit is included in this release in the developers directory.

1.13 DFBPrefs

DFBPrefs is the main front-end to DFB, from here you can edit most of DFB's features, DFBPrefs also enables you to jump to either the user editor, section editor or archivers' editor.

Gadgets

File List Layout, opt layout 1,2 & 3:

Here you can select which layout files you want to be assigned to the default layout and to the optional layouts 1,2 & 3. For an explanation of what the file layouts are see the section on installing ReadDFB There are several layout files included in this release (more will come):

standard.lay	- Default layout, this is customized for use with file_id.dizs.
longfileid.lay	- Same as the default layout but allows up to 20 line file_id.dizs (new for V1.1)
longname.lay	- This is also customized for file_id.dizs but does not display number of downloads thus allowing a longer file name.
standardshort.lay	- This layout can be used for either file_id.dizs or not. It is the same as longname.lay but only allows two description lines (of 44 across).
shortnouploader.lay	- Same as the above one but never displays the uploader's name/alias thus allowing file nodes to take up only 1 line if necessary (new for V1.11)
longdesc.lay	- Same as shortnouploader.lay but does not display date, instead it uses the room for a longer file description (new for V1.11).
nofileid_2lines.lay	- Customized for BBS's not using file_id.dizs this one allows 2 description lines.
nofileid_4lines.lay	- Same as the above but with 4 description lines.

If there is a specific one you require ask and I may be able to quickly make it for you but more than likely an editor will become available.

Description Text Editor:

When using the sysop file editor from ReadDFB you can select which text editor you wish to use to edit the file descriptions. You can select any text editor (or even a word processor if you really wanted to), I recommend diz-ed by nocturne design, this is especially made to edit file descriptions and as a plus can open up on MAX'S screen.

WBToFront:

Unfortunately when selecting a text editor to edit the file descriptions you may find that the one you wish to use refuses to open up on MAX's screen.
If this happens select this flag and DFB will automatically move to the Workbench screen - allowing you to edit your file description - and when you have finished it will automatically move back to MAX's screen.

NEW for V1.3: If you are running MaxsPatch, MAX's screens are public, just set the public screen argument to MAXS.X where X is the node number, try:

amigaguide dfb:dfb.guide PUBSCREEN MAXS.1

New Files Public:

Same as MAX's option, do you want all files uploaded to be automatically public? Any user with their access over the 'edit file' access level will automatically have their files made public. See the
archiver's editor
for more information on user access levels.

Delete on Purge:

When you delete a file using DFB, e.g., move the cursor over the file and press the 'k' key, the file does not have to be deleted then.

You may find it useful to have files deleted when you 'purge' the file base, (Purging the file base is basically optimization and removing of 'old' data) rather than immediately. If you select this you will find that pressing the 'k' key over files you have deleted they will come back! In other words this flag is useful for preventing accidents.

Use headers:

If this is selected ReadDFB will try and load the relevant header file to display. (V1.1)

FM Frequency:

See
the File Manager
Ratio Type:

This can only be changed at installation time, DFB supports three different types of file ratios:

File: This is the same as MAX's file ratio, uploads and downloads are counted in number of files.

Byte: Instead of keeping track of how many files have been uploaded, DFB counts uploads and downloads in bytes.

Credit: Is a subset of byte ratio, users only get file 'credits' or uploads as they are called when a user downloads a file they uploaded. This is useful if you find users are uploading rubbish to your system.

```
Use File_IDs:

As it says.

Ratios 1&2 Linked, Ratios 2&3 Linked, Ratios 1&3 Linked:
               See file ratios
                Enhanced->Standard:
               See user manager
                The remainder of the gadgets are used for the user manager but  ↩
                   will be
explained here.

The first column is for the 'guest' user or when 'standard membership'
expires.

The second is for 'standard user', i.e., what happens when you press
'standard member' button in the
               user editor
                The last is for 'enhanced user', similar to the above.

The first three rows are for the
               file ratios
                The next is for the users' access, then time limit and finally  ↩
                   how long
the account lasts for. Pretty basic stuff.

Menus

Only two of them need any explanation:

Purge File Base:

As discussed earlier in this section, this optimizes the file base and
does some basic housekeeping stuff. This *doesn't* perform any file
managing, but it is run automatically when you do file manage.

The file base can be purged by using the command line option:

DFBPrefs purge


Run File Manager:

This runs the File Manager, now. You should have the
               File Manager
                run
automatically.
```

1.14 DFBSection

```
                The section editor for DFB is extremely simple to use and most if ↩
                    not all
of the gadgets are self-explanatory, an overview of the most interesting
ones is given here.
```

Gadgets

Section name and number are the same as MAX's. Rules are no two sections can have the same number and try not to use section numbers above 65,000 since they may be reserved in future versions.

Upload path is very useful. MAX's BBS does support multiple file paths but DFB goes one step further, it allows each section to have its own upload path. This is merely the default path, when checking for the file DFB will check all the other paths (DFB uses the same method of paths as MAX'S does - but a word of warning if you change the paths DFB will not recognize the change until reset. This is due to its library which loads up a copy of the paths when it's needed and every DFB programme uses that one copy to save memory.)

Also when the
File Manager
moves files to another section
it will also physically move the file to the new directory (if needed). But merely moving the files to another section by yourself using a file editor will not physically move the file.

User Access to:
Access: User must have this access to list files in this section.
Download: As it says.
Upload: '' '' ''
View Hidden: User must have this access to view new files (non public files), to view deleted views and also to view the uploader's real name if the section is for instance alias only.

Edit File: As it says.
View Text: User must have this access to view files.

Use Ratio:
See ratios
Flags:

Section Off-line:
By clicking on this check box you can take the section off-line. The files will remain the same but users cannot access them. Clicking on the check box again will put the section back on-line. This is useful if say one of your HDs is being repaired and you still want users to know your board has the files.

Section Free: By clicking on this check box you make all files in that file section 'free downloads'. So any user can download them without affecting his/her credit ratio. Any file uploaded to that section will then become a 'free download'. By clicking again on this check box files go back to their previous state (whether they were free or not). (V1.2)

Uploader Signatures:
Set if you will allow upload signatures to be used.

Cycle gadget, Uploader:
What to print when DFB prints the uploader.

This allows you to use the same layout for different
sections and yet customize it down further. MAX'S has
support for several:
Alias sections
Sections with only the users real name
Section with no name or alias.
DFB allows tons of combinations. Just try them :-)

This is not connected to upload signatures in any way so
you could have both or neither of them running.

Max Size/File Life/Then Move to Section:

See File Manager
Menus

Clone Section: This is used to add sections, to add a section first go
to a section with similar attributes (e.g.: access levels
etc.) then select clone section from the menu - a new
section with the same attributes as the section you were
last on, will be created. The only differences being the
section name and number.

The remainder are self-explanatory.

1.15 DFBSection - File Manager

In MAX's BBS, when a file is uploaded to a file section and the ↩
maximum
number of files in that section has already been reached MAX's will delete
the oldest (non-locked) file.

This is MAX's File Manager. DFB goes several steps further.

You cannot set the maximum number of files in a file section, I fail to
see the point of this option. Instead you can set the maximum
size of the file section in megabytes. Also another difference is
DFB does all its file managing at the end of the day - so if a user uploads
a 2.3 meg file to a file section and that section is now over its megabyte
limit the oldest files will not be deleted until the File Manager is run.

There are 3 main gadgets which control the File Manager and how it
operates, they are found in each file section in the section editor.

The first is Max Size, this controls the size in megabytes of the file
section in question. If Max Size is set to zero DFB considers the file
section to be infinite in size.

File Life, if you want files to remain in a certain section for a limited
period of time set the number of days here (+1). If you set this to 0,
the File Manager will only move/delete files if the maximum size limit
is reached.

The Move to section, if the maximum size of the file area in megabytes

is exceeded or the life of a file is exceeded (that is if you are using
the file life gadget) then this gadget controls where the files go to.
If it is set to 0 the files are deleted otherwise the files are moved to
that section.

Note: You cannot move a file to section 0 with the File Manager. If this is
a problem don't use section 0, there are, plenty of others to choose from :-)

Running the File Manager

*Before you run the File Manager make SURE every section is set up
 correctly. The File Manager is powerful and does delete files, as you
 can imagine misuse could have disastrous consequences*

Running the File Manager is similar to the
               user manager
                The File Manager has to be run to keep order and perform ↩
                   automatic
housekeeping tasks. It can be run once per day or once per week depending
on the amount of traffic your BBS has.

To run it, either select the option 'Run File Manager' from the
               DFBPrefs
                menu or type from CLI:

DFBPrefs filemanage

Suggested ways of running it:

1) Put the above line in your startup-sequence, DFBPrefs will detect if it
   needs to be run so frequent computer resets won't cause it a problem.

   If you choose this there is a gadget in DFBPrefs called FM Frequency
   which dictates how many days apart the File Manager should be run.
   This way you can have DFBPrefs running the File Manager in your
   startup-sequence and it can still run every x days. An FM Frequency
   of 0 means the File Manager is run everyday. FM Freq of 1, the file
   manager is run every other day etc.

2) Select the menu option 'Run File Manager' in DFBPrefs every now and
   again :-)

3) Use MAXsBBS AutoExecute command.

1.16 DFBSection - File Ratios

It would be ridiculous to release a file base system with no file ratios
because every sysop expects it, DFB goes one step further, for every
user there isn't just one file ratio there are three.

Why three ratios?

I admit, most users will probably only require one file ratio. Having
multiple file ratios gets a bit complicated, especially for the users.

When you install you'll see that only the first ratio box is used. Using more than one file ratio is optional.

The reason being is that you might have files on your HD which are say newer than others and you find it unfair that users can say upload some old programmes and then go and download heaps of the newer programmes.

This way you can have one ratio for all your old programmes, and if a user wants some newer ones he has to upload to the new section. In this example you could also use the File Manager to automatically move the files after say 30 days when the files are no longer new.

Originally the ratios were going to be linked, so even if the user didn't upload the 'new' files he could still download them - it just meant some file sections required more file credits than others. This 'linking' uses the three disabled check boxes in the DFBPrefs programme and will probably not be done. If you require this feature mail me!

1.17 ReadDFB

ReadDFB is the only programme which is run from MAX's BBS. This programme can emulate the MAX's functions: upload, download, list files, list new files etc.

Firstly ReadDFB must be installed into MAX'S BBS, this process is not automatic because I thought people wouldn't appreciate me just changing their whole board over to my file base without them knowing :-)

Select what you wish to know about ReadDFB:

Installation

User functions

Sysop functions

Customization

1.18 ReadDFB - How to install

How to install

Enter MAX's menu editor and change each line which accessed MAX's file base to equal:

Key:	Function:	Extra:	Lo acc:	Hi acc:	Filename/Name/Dest/Path:
?	34	0	?	?	DFB:ReadDFB <options>

```
Where the question marks remain what they were originally.

Now there are several options which ReadDFB can use, the options are
split down into several groups which have to be in this order:

(If no options are given ReadDFB will list every single file using default
 options)

*All the important options are marked with a star.*

ReadDFB [<layout>] [<type>] [<command>] [R] [<section lists>]
        [<(upload section)>][<"headername">]

<layout>:  DFB can display the files in a variety of ways. e.g.: Should
           DFB display how many times a file has been downloaded? How
           many lines should be given for the description? You can specify
           which layout you want to use, either 1,2 or 3. This number
           matches the layout selected in DFBPrefs.

           e.g.:

           ReadDFB L1

           Will select the first optional layout.

  <type>:  You can list files using certain criteria such as list new files,
           list off-line files, list new files etc.:

           D    List deleted files
           L    List locked files
           N    List new files
           O    List off-line files
           +    List free downloads

           Put a '|' in-between them, e.g.:

           ReadDFB D|L|+

           will list files which have either been deleted, locked or are
           free downloads.
          *<command>: THE most important option, this selects what you want
ReadDFB
           to do. If you do not select a command ReadDFB just lists the
           files.

           The commands are:

           down       (Just download files, don't list them)
           up         (Just upload files,    ''    ''  '')
           last       (List new files since last on (like MAX'S))
           tag        (Only list tagged files)
           find       (Find a file by text search of both name & description)
           qfind      (Find a file by text search of just the name)
           date       (List files after a date (or before if you have
                       selected to list files in reverse order, see later))
```

[R]: Direction, if you put in an 'R' DFB will list the files from newest->oldest.

*<section lists>:
When you are listing files it's essential to choose which sections you want listed at a time.

Best way to explain is to give some examples:

ReadDFB	List all files
ReadDFB 1	List only the files in section 1
ReadDFB 1-100	List only the files in sections 1 to 100 (inclusive)
ReadDFB 1,3	List files in section 1 and files in section 3
ReadDFB 1,1-20,1000-1012	Guess :-)

Note: Don't get too carried away...

Some points: The sections need not exist, e.g. telling ReadDFB to list all files in sections 1 to 100 and not having any sections between 20 and 30 will not cause a problem.

The user will only see the files in the sections he has access to. So feel safe to group sysop files with ordinary files, only the right guys will see them or be able to access them :-)

<upload section>:
If only one file section is given DFB will naturally assume that uploads go there but it can get a bit confusing if you have a general file list.

So if you are listing files from multiple sections and you want the user to be able to upload just put the section you want the uploads to go to in brackets.

e.g.:

ReadDFB 1-1000 (20)

List files in first thousand sections, uploads go to section 20.

Or, if you have only listed one section but you want the users' uploads to say go to a different section (perhaps a new files section) then:

ReadDFB 3 (4)

Will list files in section 3 but all files uploaded will go to section 4.

"<headername>":
One of version 1.1's new features. Header files.
See

```
                Headers
                 for more information.

            Forces ReadDFB to load a specific header file.

            e.g.:  ReadDFB 3 (12) "DFB:headers/privateheader"

            (You must include the quote marks).

Feel free to combine all of the above but don't get too carried away.

Here is a couple of examples:

ReadDFB Last 1-100 (20)       List new files since I was last on in first
                              one hundred sections. Uploads go to 20.

ReadDFB N R                   Sysop one, list non-public files (e.g.: new
                              uploads) starting from the latest one.


One more thing, DFB has on-line help by pressing the 'H' key. ReadDFB then
displays the text file: DFB:DFBhelp.txt which can easily be edited.
```

1.19 ReadDFB - User functions

```
                DFB is simplicity itself to use so I'll only briefly cover the ↩
                   options:

The bottom line gives most of the user commands:

-> Arrows to select, U)p D)own R)everse H)elp S)top <RETURN> -->
^
|- Direction of file list, towards the right is from oldest files to newest.

Up and down arrows to select files, <RETURN> from the bottom line will
continue the file list.

U) Upload files
D) Download tagged files and/or other files which user can enter in.
R) Reverse the direction of file listing starting from files which aren't
   visible.
H) Help, views text file DFB:DFBhelp.txt
   (Thanks go to Bonza for writing the included one)

S) Stop listing

Not shown:

E) Edit upload signature, if upload signatures are used the user can
   edit his/hers. The user has a choice of auto-centreing it to make life
   easier.

I) Gives the user information on how many files uploaded/downloaded, how
   many to go and ratio. (For byte or credit shows the number of bytes)
```

If the user moves the cursor up:

<RETURN> to tag/untag, D)own V)iew R)everse S)top

<RETURN> tags/untags files 8-)

D,R and S work as above.

V) View files, see
Customization
Ctrl-L will redraw the file list.

At the end of every file list the user will see:

Files listed/total: [xxxx/yyyy]

I think this is self-explanatory.

1.20 ReadDFB - Sysop functions

The access needed for a user to have sysop functions is given in the
section editor. Each section can have a different level for sysop functions,
so a user might be sysop for one file section but not another.

These sysop functions can still be performed on an account which only
has standard user level ONLY if the key press came locally. I.e
if you are sitting down at your computer and you notice that
a file listing looks weird when one of your users lists it, you can edit
the file using that person's ReadDFB, even if that user doesn't have high
enough access.

ReadDFB will note that the key press came locally and will let you
edit the file.

Editing files

Move the cursor over the file as though you were about to tag it.
The following keys will toggle the files attributes:

K Delete file, if there is a 'D' against the file it has been deleted,
The 'K' key will delete a file and can bring the file back _only_
if you have selected 'Delete on Purge' check in
DFBPrefs
N New file, if there is an 'N' against the file it is not a public file.
The N key toggles this. Only sysops can see files which are not
public (unlike MAX'S).

L Locked file, the same as MAX'S BBS. The L key toggles this.

O Off-line file, in DFB you can include files which are not on your
HD. With off-line files users can see which files you have but not
be able to view/download them. Also if you are temporarily unsure

about a files contents you can put it off-line, users can still see you have it while only sysops can view/download it.

+ Free downloads, if a file is tagged with this symbol users can download it even if they have a lousy file ratio. They will not be charged :)

E If this letter appears next to the file name it means the file is waiting for the uploader to type/retype in a file description. To toggle this flag, press 'e' to enter the file editor and then toggle the 'NeedsDesc' flag.

If you want to edit something else, say the name or description of a file press 'e' over the file.

A window will come up with all the file's attributes. Most of the gadgets are self-explanatory so I'll only describe the interesting ones.

The string gadget next to the section number contains the name of the section the file is in. If you want to move the file to a certain section, type in the section name or a part of the section name and the window will jump to the correct section.

For example, if you want to move to a section named 'Comm & Terminal Progs', you could enter in the section text gadget 'term' and it would move the file to the right section.

There is an additional checkbox available in the local file editor window, it is labelled NeedsDesc. If this checkbox is checked the user who uploaded the file, will be asked when they log in to leave a file description. Or DFB will search for the file's file_id.diz if enabled upon user login.

There are four buttons down at the buttom of the window:

Description - DFB will load your own custom text editor to edit the file's description. You can choose your text editor with DFBPrefs. I *totally* recommend Nocturne Designs's diz-ed. This is absolutely brilliant and it can open up on MAX's window. BTW just so DFB can use the above editor you may see: @BEGIN_FILE_ID.DIZ & @END_FILE_ID.DIZ, make sure the description is in-between these.

Nuke Credits - Don't like the file that you're editing? This button removes the uploader credits he would have gained from this file. Multiple clicks will severely annoy the uploader :-) If the screen flashes it is either due to: 1) Can't find uploader
or 2) User has no more credits left.

Zot File - If you really want this file gone, Zot it! This will immediately delete the file and will remove all mention of the file in any ReadDFB list.

Save - This will save any changes you have made to the file.

A remote sysop can also edit files, the procedure is the same. Move the cursor over the filename and press 'e'. A new screen will come up with all the options. Currently file descriptions cannot be edited from remote.

1.21 ReadDFB - Customization

Most of ReadDFB's customization is covered in the Installation bit of this document although a major part is what happens when a user tries to view a file.

This is fully configurable by you, DFBArchivers contains the relevant information on how to set it up. See DFBArchivers - viewing files

Another customization feature which is new for version 1.1 is Headers, this is fully explained in its own section.

1.22 ReadDFB - Headers

What are headers?

Headers are small text files which tell ReadDFB how to display its information. Currently when listing files you can change the colours of: filename, date, times uploaded, description, uploader, alias, size and filename when highlighted. You can also make ReadDFB display ansi texts at the start of file lists.

How do I use them?

First of all you must enable them, go to DFBPrefs and make sure that the headers check box is selected.

Now ReadDFB is ready to use headers! First of all ReadDFB has its own system for working out which header to use. Each ReadDFB command can load a different header so every section may have a different header and combined file sections.

ReadDFB will now choose a default header to load. It's a pretty simple system, if you are listing only the files in section 1 it will load the header file: DFB:headers/header.1 If you are listing files only in section 3 it will load the header file: DFB:headers/header.3

If you are listing combined sections it will default to the file:

"DFB:headers/header." (with the period mark)

Now this is a bit limiting, since you may wish to put your header files
in RAM: to speed things up or you may wish to have several different
headers for all your combined file lists.

You can force ReadDFB to load a specific header file by placing the
header file name in quote marks as the last command line argument,
e.g.:

ReadDFB 100 (12) "ram:myheader"

How do I modify them?

This is the tricky part, DFB's philosophy has been to keep it simple,
easy to use and as powerful as this will allow. Unfortunately I couldn't
think of an easy way to use the power of headers and have a nice GUI
for editing them. (It would be too large!). There is a demo header file
included in this archive (DFB:headers/header) and by following that
it's very simple to edit to the configuration you want.

I will go over the rules briefly here but don't worry! There is no
reason why you can't just modify the included one.

Here's the rules in case you need them:

1) If you want a comment in the header file put a '#' as the first
   character.
2) If you wish to use one of the commands (they are used to change
   colours, put a '$' in front of it)
3) Don't use blank lines except in the text bit to display
4) If you want to use the '$' or '#' symbol as the first character in
   your ansi text to display write two of them in the text file. ReadDFB
   will know what to do.
5) All commands should be in lower case.

OK now feel free to forget them! Just go and edit the included one!
All you need to know is comments and commands go at the top, the ansi
text to display goes down the bottom and doesn't have blank lines in-between
comments and commands.

One more thing, you can only choose the colours: white, yellow, green,
red, cyan, purple and blue. Notice no black!

Have fun!

1.23 DFBArchivers

The programme DFBArchivers controls several settings of DFB.

Archivers, there are 4 columns in the archivers' editor, the first is the
extension. This covers the extensions which files can have, e.g.:
.txt, .lha, .g etc.

You can have up to 11 characters in here, so if the same commands should
be executed for say .lzh files and .lha files you can put multiple
extensions in separated by a comma:

lha,lzh

The other three columns control what happens when DFB encounters these
files at certain points:

```
               Extractor for file_id
                       - What command DFB runs to extract the
                                    file_id.diz (to the current directory)

               Command On Upload
                       - What command DFB runs when a file
                                    is uploaded.

               File List
                       - What command DFB uses to make some
                                    text to put on screen when a user tries

                                    to view a file on-line via ReadDFB
```

The archivers' dir gadget is where DFB will look for your archiving
programmes.

You may think this is fine for programmes which end in extensions, but some
files (mainly text ones) have no extension or what if DFB encounters a
file with an unusual extension? (such as .bmap)

For this I have added the 'default gadget', you may notice that along side
each row in DFBarchivers is a number. By placing this number in the default
gadget it tells DFB which row to use if it encounters a file with either
no extension or an unrecognized one.

The default number is 0 - which means don't do anything.

Also DFBArchivers controls
               File backups

1.24 DFBArchivers - File backups

When the File Manager is run, DFBArchivers, as a safety precaution backs
up the following files: DFB.prefs, DFB.users & DFB.data.

You can backup/restore those files any time by using the menu options in
DFBArchivers.

The file to which these are backed up is given in the gadget in DFBArchivers,
if this string gadget is empty DFB will not back up the files.

The following command is always used to back up files:

lha >NIL: u DFBdata:DFB.data DFBPrefs:DFB.prefs DFBusers:DFB.users

to restore them:

lha >NIL: e DFBdata:DFB.data DFBPrefs:DFB.prefs DFBusers:DFB.users

Also make sure you have specified the directory your archivers are in!

Note: DFB handles file paths like MAX'S BBS, for simplicity, so

hd0:a/ is acceptable whilst:
hd0:a is not.

1.25 DFBLogin

Installation

To install DFBLogin is easy, just copy the file DFBLogin to:
BBS:Doors/DFBLogin
Then add the line:

DFBLogin

to the file BBS:Doors/IntroDoor.text

What it does

DFBLogin checks the user's status to see whether the user uploaded file(s) without leaving file descriptions. If the user did it will then ask the user to enter the file descriptions. It will then update the files in the file base.

It also turns the user's pause flag on. This is necessary since due to a MAX's problem, if a user loses carrier in DFB their pause flag will be turned off.

What if I want DFBLogin to run AFTER mail?

If for some reason you would prefer DFBLogin to be run AFTER mail then put the line:

DFBLogin

in the file BBS:doors/MainDoor.text
AND move both DFBLogin and pauseflagon to BBS:Doors/

and put the line

setpauseflagon

in the file BBS:doors/LoginDoor.text

This is necessary so that the user's pause flag is on BEFORE mail is run.

1.26 DFBArchivers - File_Id extractors

If you have the File_id.diz flag off in DFBPrefs this column will be ignored.

DFB contains no code of its own to extract file_id.diz's, instead it relies on external files.

The format is relatively simple, just write in the command that you would use to extract the file_id.diz. Where the file name would normally be kept replace it with a '<>' which DFB replaces with the file name later.

e.g.:

```
lha >NIL: x <> file_id.diz
unzip >NIL: -e -o <> file_id.diz
```

Will extract file_id.diz's from the files with the extensions which would be specified in the extension box.

1.27 DFBArchivers - Viewing files

Viewing text files is handled slightly different from the

file_id.diz extractors
&
command on uploads
How it works

When a user selects 'view file' DFB will immediately check the file's extension. (e.g. .txt etc.). If one is not found DFB will say it can't view the file, if it does find an extension it will go to the appropriate 'File List' box.

It will then replace the '<>' in the command with the full path and name of the file DFB wants to view. It will then execute what is in the box.

After that DFB will check the current directory for a file called ListX, if this file is found DFB will print it out on MAX'S BBS's screen.

Example

Say when a user tries to view an 'lha' file, I wanted DFB to print out the files in the archive, this is the command I would enter into the box:

```
lha >ListX l <>
```

Now, the first > symbol will redirect lha's output to the file ListX in the current directory which DFB will then read.

What if it was a plain .txt file?

Well there are two ways of doing it, a:

```
copy <> ListX

Would copy the file to ListX and DFB would then read it but since text
files are so common I included the operative: ! which is used to display
a text file, so the above command can be written:

!

The ! operative is handled internally and is better than the copy command
because it tells DFB just to read the file straight from disk.
```

1.28 DFBArchivers - What to do when a file is uploaded

```
                If you want DFB to say add in a BBS ad or to remove someone else' ↩
                   s <grin>
this is where you put the command, e.g.: if you wanted to put a file
called 'mybbs' which is kept in ram: to every .lha file uploaded you'd
put in the 'Command on upload' box the following command:

lha >NIL: a <> ram:mybbs

Like the
               file_id.diz box
                the <> sign indicates where DFB should put in the
file name.

Note: If you want more than one command to be executed you must create a
dos script file. Remember that any command put in the box will be prefixed
with the archiver's directory.

Hint: When using dos scripts try this: Put the dos script into your
archivers directory then type from CLI:

protect dosscript +s

Then all you have to put in the 'command on upload' box is:

dosscript <>

The '<>' sign passes on the file name being processed to the dos script,
you must do this, see any dos manual for how to pass variables to a script.
```

1.29 DFBUser

```
                Due to some new features of DFB the standard MAX's user.data file ↩
                    had to
be expanded, all the new information just wouldn't fit into it, and even
if it did, MAX'S does not have the features to edit it anyway so it was
necessary to build another user editor.

So at the moment you have two different user editors! The MAX's one and
DFB's. At the moment DFB's user editor only contains relevant information
```

which DFB needs (hopefully in a future release it will be a full editor).

Here are the relevant sub-topics:

When to use DFB's editor, when to use MAX's
important

What everything does

The User Manager

1.30 DFBUser - When to user DFB's editor, when to user MAX's

If you are editing a user's access or his time limit then either editor will do. If you use MAX's the change may not be immediately visible in the DFB user editor, however the next time the user logs on DFB will act on the new access level (or time limit).

If you are editing the user's ratio, number of files (or bytes) uploaded (or downloaded) then you must use DFB's user editor. After installation DFB ignores the MAX's values for various reasons.

If you are changing a user's name for some reason change it using the MAX'S's user editor then go to DFB's editor and change the name there too.

If your user.data file changes for some reason, e.g. a new user logs in or you delete a user or whatever DFB will gradually change its user file when it needs to, if however you want to view the changes immediately select the 'Update Users' from the menu.

1.31 DFBUser - What everything does

Gadgets

Most of the gadgets, if not all, in the user editor are pretty easy to guess. Here is just a brief mention of some of them:

User Name: If you have to edit the name make sure you edit it in both MAX'S's and DFB's user editors to prevent confusion.

Upload Signature:
New feature for DFB. An alternative system for uploader's tagging their uploads. At install time each section has upload signatures disabled.

Days Left:
New for MAX's - DFB has an built-in membership door and this is the main part of it. Each user can have an account with a set number of days before it expires. The number of days can be set with the standard/enhanced membership buttons.

Standard Member/Enhanced Member:

```
               Similar to MAX'S membership buttons. The options for what
               happens when you select these buttons are given in
               DFBPrefs.

          Save:
               Saves changes to user.
          Next:
               Go to next user.

As you may note there are three boxes for uploads, downloads and ratios.
There is an explanation for this in the section on
                file ratios
                 Menu Items

Update Users: This performs some basic housekeeping functions, this is
               automatically run when you use the user manager.

Run User Manager:
               This runs the
                user manager
```

1.32 DFBUser - User Manager

```
Firstly...

DFB has a built in membership door, at the moment it is very simple and
only supports the following:

Quick update buttons for 2 different access levels (Standard & Enhanced)
Which updates:
 * Time limit
 * Days of account
 * Ratios

These are set in the DFBPrefs programme.

The User Manager

The User Manager is used so DFB can keep track of how many days are
remaining in everybody's account - if you do not have any accounts which
expire or change after a certain amount of days you do not need to run
this!

The User Manager should be run once and only once per day, every time it
is run it first checks whether the users need updating, then it subtracts
x number of days from everybody's account (accounts with 0 days left are
considered to be everlasting) if an account reaches 0 it is then reduced.

A standard membership account will expire to a guest account (i.e., the
first column in the DFBPrefs account gadgets)

An enhanced membership account will either expire to a standard one or a
guest account depending on whether the tick is set in DFBPrefs.
(The Enhanced->Standard check box)
```

When I say x number of days this means if the user manager detects it hasn't been run in say 3 days it will subtract 3 days from everybody's accounts.

Recommended ways of running the user manager:

1) Put the following line in your startup-sequence:

   DFB:DFBuser usermanage

   DFBUser detects if it has already been run on any given day so if you reset your machine several times it will not keep on subtracting 1 from the days left in everybody's accounts :)

2) Use MAX'S BBS's AutoExecute command.

Running the menu option is not recommended since it _always_ subtracts at least one from the number of days in everybody's account. So use it wisely.

1.33 Feature List

These features are optional so if you don't want them don't worry!

* 100% Assembly coded for minimum size, maximum speed and power!
* Programme automagically converts from MAXS -> DFB. No messing around with text files to get this programme working
* GUI for preferences selection
* Built in remote file editor
* Full file_id.diz support (for anything, e.g.: .lha, .zip etc.)
* Filenote support
* Automatic 'file ad' addition (execute any command on upload)
* Can view almost any type of file on-line: (e.g. .pp, .guide are no prob)
* Conference ratio support
* Support for the following ratio systems: file, byte or credit.
* Log off after downloads
* Up to 65,000+ file sections possible leaving all 100 MAX'S sections free for mail
* Uploader signatures and uploader aliases
* Local uploader uses an asl requester with multiple file selection
* File & Sections can be taken off-line with a click of a button (Full offline file & section support)
* File & Sections can become free downloads with a click of a button
* MAX's style file tagging (even when using file_id.dizs!)
* Make files public, lock files, make files free dls, move files off-line, delete files by just moving the cursor over the name and pressing a key!
* Each section has its own access levels for: downloading, uploading, viewing, sysop functions etc.
* Each user has his/er own tagged file list, doesn't 'disappear' if they lose carrier (tagged file lists can be kept anywhere and sysop can edit it with any text editor)
* List *any* combination of file sections at a time: e.g. list sections 1,2,3, 23 and 300 to a thousand (inclusive). If a user doesn't have access to one of those sections he won't see the files in that section.
* List files using almost any possible criteria: By date, by name by name and description, tagged files, list only off-line files, list

off-line files or new files etc. etc... combine this with the listing of any combination of sections.. !
* Possible to change the way DFB displays files. Don't want it to display number of downloads? No prob. Rather use only 3 description lines? Just use a different layout!
* File headers, which allow text to be displayed at the top of every file list and also allows you to change the colours of the file list
* User can change the direction of the file listing at any time (THIS FEATURE IS UNIQUE TO DFB!)
* File Manager: Automagically move/delete files after x days or move/delete files if section is full
* Automatic back-up of file base etc.
* Each section has its own default upload directory
* Set each section size in megabytes
* User manager and membership door: Give users access at the touch of a button, define the time limit, membership duration, ratios etc...

1.34 Credits & Greetings

Main coding/design by: Andrew Leppard

Initial idea, concept and beta testing by: Bonza
additional code donated by: Randy Quick
additional coding help by: Michael Zucchi

Thanks go to: Randy Quick & Ari Tsironis for information concerning MAX'S BBS door programming.
Bonza for not giving up at the sight of those _rare_ bugs :)
Jan van der Baard and everyone at Jaba Development for GadToolsBox which was used to make all of the GUIs. It's a brilliant programme!

Micke Lehner, Shaun Payne, Randy Quick & Egress-1 for allowing their boards to be used as distribution sites.

Thanks to my new distributors: Ray De Brabander & Paul Bradley.

Greg Fitch for alerting me promptly about bugs!

Thanks to The BenchMaster for ideas and spell checking the manual.

Thanks to the following people for beta testing V1.2:

Bonza, The BenchMaster, George Kalaitzis, Tommy Samuelsson.

Thanks to Heddley author Edd Dumbill for his excellent program and to FixHeddley author David Zvekic.

All those that have stuck with MAX'S BBS this far!

Greetings to: Mav (why didn't you do the GUI for me?!@!@!), Zed and everyone else at Frontier (OK, OK so I should have done some code for

you?), Randy Quick (when's the 5th line coming? :), Cydonia, Pearl and Digital Access, Angela Borchard, Spoonman, Bonza (what do you mean uploads don't work?), The Sorcerer, Wyrehead (you could have told me about that Amiga IRC session you were on!), The Haggis, Dizzy, Ari Tsironis (Pity about MAXSBBSTNG - there _was_ a big market for it), a1, Azat, Cro, Prowler, Sal, Jody Christine & Samantha Murphy, Benchie, Melissa Page.

The Jaker, Blackout, AITEC and anyone else whom I may have forgotten. Big wave to Chastity from Chastity Records.

Appreciation to/of:
Bomb the Bass, The Prodigy, Culture Beat, Urban Cookie Collective, Jolt Cola, Heaven the club, MAX'S BBS, Apotheosis (for Obumbrata), JX for 'Son of a Gun' and 'You belong to me', Amiga and all that have worked on it, Suns and the Internet, -- S-Express --, Itchy & Scratchy!

No-Thanks go to:
ACE-TV, S.A Police force (for being there), IBM (R&D exempted), Microsoft (NO EXEMPTION), Reggae, Anthony Barret (for not helping MAX'S BBS programmers now), Country & Western and some guy called Bill and another called Drew. Stay out of the park Bill.

And finally this programme is dedicated to Jody Murphy, because if you can't dedicate a programme to someone, what can you do?

1.35 Future of DFB

Firstly, DFB is not perfect. As far as BBS file bases go, it is getting there though! So what do I plan to do to DFB in the future?

* Utilities, doors e.g.: Top Up/Downloaders.
  (Someone started programming this!)
* XPR transfer protocol libraries. This way DFB will support more protocols such as Super Zmodem, Bimodem etc... (already done most of the work)
* Daily download limits
* CDROM/DAT section support (including multiple DFB.data file support)
* A local file editor launched from DFBPrefs with some serious searching options. The DFBCheck one only works on files with problems!
* Lots of minor enhancements
* Can't think of anything else. All of the features I could think of are already in it!

1.36 How to Register

Why register?

Well, quite simply, the demo (unregistered) version of DFB is next to useless, downloads have been disabled to encourage you to register.
Also the saving of tagged files to disk has been disabled.

How to register

First of all, fill out the
Registration Form
, this contains important
questions so come update time I know what needs to be done, because at this time I am not a sysop and don't know exactly what you need!

Either...

Post it to my internet account:

E-mail: 9405571x@lux.levels.unisa.edu.au
If that one *DOESNT* work then try: a.leppard@aitec.edu.au

Or post it to me in a message on Fred's BBS:
Line 1: +618 341-5944 (2400 baud)
Line 2: +618 341-5137 (14.4k baud)

Or leave the registration form on one of DFB's distribution sites mailed to the sysop. (On Fred's BBS mail it to me)

Or if all else fails, snail mail to:
11 Gordon Tce,
Rosslyn Pk
5072,
South Australia

Finally if you _really_ need to talk to me my phone number is:
+61(0)8 8332-0002 (Voice) but please no MAX'S problems! (Unless they are directly related to DFB).

Once you have done this you will need to pay the small shareware fee of $30 Australian before you receive the registered keyfile! Suggested methods are a postal money order or a cheque for $30 Australian.

If you live in one of the following areas you will receive your copy of DFB from the nearest distribution site:

Europe (except UK):

Ray De Brabander
Amiga Force BBS EHQ
+31-0-70-3604728
Maxsnet: 86:20042/200

UK:

Alex Jenkins
The Cremlin
+44 (0)121 321-3449

USA/Canada:

The BenchMaster
The Bench//\/\aster's BBS

(408) 238-5885
San Jose, California

I'm looking for an East Coast USA and a Canadian distribution site. What do you need? An Internet account and a MAX'S board running DFB, and must live in the USA or Canada (but not both :).

Maybe even a South American distribution site will go down well.
Convince me. 8-)

Australasia:

Andrew Leppard, USER of
Fred's BBS
+61(0)8 3415944 (2400)
+61(0)8 3415137 (14.4k)
Maxsnet: 86:308/111

Then I will send out the latest registered version of DFB, the only registered file is a keyfile. All DFB programmes will recognize that you have registered and praise you accordingly.

The method of transfer will be discussed. If you posted the registration form on one of the distribution sites you can get your copy off of there.

Registration fees cover all updates up to (but not including) version 2, these updates will be available on Aminet and all the distribution sites, registered versions will be sent out upon request.

1.37 Congratulations

You have found the hidden link in this AmigaGuide document. Now find the hidden text messages in DFB! 8-) <you'll need to have registered>

1.38 Registration Form

A copy of this form is available separately...
Just Print out this form and mail it to me.

REGISTRATION FORM FOR DFB (DYNAMIC FILE BASE V1.3)

DFB V1.3 costs $30 Australian, this includes free updates. Upon payment you will receive a registered keyfile.

All information in this form is confidential.

Name:
~~~~~~~~~~~~~~~~~~~~~~~~~~~
Address:
~~~~~~

~~~
~~~

~~~~~~

~~~~~~

~~~~~~

Put a cross next to any of the following if you are interested in them:

( ) New message base system for MAX'S
( ) New menus system for MAX'S
( ) Utility to allow CLI programmes to be run as MAX'S doors
( ) Utility to allow AREXX doors to be made for MAX'S.

( ) Other

~~~~~~

Which distribution site do you want to pick up your copy of DFB:

( ) Europe (Amiga Force BBS EHQ/Ray De Brabander)
( ) U.K (The Cremlin/Alex Jenkins)
( ) U.S.A (The Bench//\/\aster's BBS/The BenchMaster)
( ) Australasia (Fred's BBS/Andrew Leppard (user))

( ) Via the internet, uuencode it to my mail account (enter above)
( ) Via airmail. Please add $5 for handling/mailing costs.