compare.module

Leo Davidson

COLLABORATORS			
	TITLE : compare.module		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Leo Davidson	February 12, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

compare.module

1.1 compare.module guide

```
              compare.module v1.2 (14th October 1996) for Directory Opus 5.5
Written by Leo 'Nudel' Davidson for Gods'Gift Utilities

A plug-in module for Directory Opus 5.5 to compare two files, report
how many differences there are, and show exactly what is different.


             Requirements

             Introduction

             Installation

             Setting Up

             Arguments

             Using
            _
             Example Full Report

             History

             Future

             Source Code & Updates

             Credits

             Purchasing Directory Opus 5.5
            The Amiga is in a perilous state right now, and, apart from the  ←
                ageing OS,
all that we have left are devoted devlopers and their innovative programs,
and a whole lot of hopes and dreams. Developers like GPSoftware keep this
platform alive by keeping it at the cutting edge with software like
Directory Opus 5.5. If you are using a pirate version of Directory Opus,
send a signal that you want there to be further versions by purchasing it.
Don't let the bridge to our dreams die. The only way you can support the
```

Amiga is by supporting Amiga Developers.

1.2 compare.module: example full report

```
Example Full Report
------------------------------ -- -  -
Please note that if you are not using the AmigaOS3 AmigaGuide datatype
this page will not be coloured and so you won't really get a good idea of
exactly what a full report shows you: the bytes which are different
should be highlighted in another colour.

In the example below a copy of the compare module was made and edited.
The word "contents" and two occurances of the word "files" were made
uppercase. Later on in the file the version-string was changed so that the
file pretends to be v9.3 when it is really only v1.1, and the date of
release has also been modified.

Apart from these changes the two files are identical, so most of them is
not shown.

If all this doesn't really sound that useful to you, I guess you haven't
been using computers for very long. <grin>

If you do not understand the Hex/ASCII dump format, read up on the HexRead
command in your Opus5.5 manual.

----- 8< ----------------------------------------------------------

 Left: RAM:Temp/compare.module
Right: RAM:Temp/compare.module_fake

00001ce0 | 6d706172 696e6720 mparing  | 6d706172 696e6720 mparing
00001ce8 | 66696c65 732e2e2e files... | 46494c45 532e2e2e FILES...
00001cf0 | 00000000 03ea0022 ......." | 00000000 03ea0022 ......."
00001cf8 | 436f6d70 61726520 Compare  | 436f6d70 61726520 Compare
00001d00 | 74686520 636f6e74 the cont | 74686520 434f4e54 the CONT
00001d08 | 656e7473 206f6620 ents of  | 454e5453 206f6620 ENTS of
00001d10 | 74776f20 66696c65 two file | 74776f20 46494c45 two FILE
00001d18 | 73000000 03eb0004 s....... | 53000000 03eb0004 S.......
00001d20 | 4f4b0000 000003ec OK...... | 4f4b0000 000003ec OK......

00002ee8 | 6f6d7061 72652e6d ompare.m | 6f6d7061 72652e6d ompare.m
00002ef0 | 6f64756c 6520312e odule 1. | 6f64756c 6520392e odule 9.
00002ef8 | 31202833 2e31302e 1 (3.10. | 33202836 2e31312e 3 (6.11.
00002f00 | 39362900 00000000 96)..... | 39392900 00000000 99).....
00002f08 | 00010000 064a0000 .....J.. | 00010000 064a0000 .....J..

----- 8< ----------------------------------------------------------
```

1.3 compare.module: requirements

```
                        Requirements
-------------------------------- -- -  -
```
This module requires Directory Opus version 5.5 —- it will not work with
version 5.11 or other earlier versions.

If you're using a pirate copy of Opus5.5, I'll pretend it's YOU I'm
shooting at in Quake and Descent 2. =8-| ...and something nasty might
just happen to either your harddisk or your genitals, depending on my
mood and the phase of the moon.

When generating
                full difference reports
                on large files you will need
quite a lot of free memory. Doing simple compares only requires around
100k of free RAM.

Apart from that there are no unusual requirements. If you have any
problems running it on your system, please
                contact me
                and I will do
my best to help out, or write to the Directory Opus 5 mailing list.

1.4   compare.module: introduction

```
Introduction:
-------------------------------- -- -  -
```
Have you ever had two files which look the same and wonderred if they
really are the same? Then compare.module may be useful.

Given two files, compare.module looks at every byte of them and tells you
how many are different. You can also get it to show you the parts of the
files which are different so you can see exactly what has changed.

Because it is specially written for Directory Opus 5.5 it is extremely
easy to install and use, and you also get bonuses like a standard progress
bar while it works.

1.5   compare.module: installation

```
Installation:
-------------------------------- -- -  -
```
Use the supplied Installer script, it won't bite.

To install by hand, simply copy the file compare.module to DOpus5:modules,
and compare.module.guide to DOpus5:help. Although you probably won't have
to, you may need to reboot or restart Opus for the module to be activated.

If you don't want to use the built-in English strings and a catalog exists
for your preferred language, copy the appropriate directory from catalogs/
to DOpus5:catalogs/.

1.6   compare.module: setting up

```
Setting Up
-------------------------------- -- -  -
Once installed, this module adds a command called Compare to Opus.

The Compare command will now be shown in the pop-up list of commands with
all the other internal commands of Opus.

You should setup a function in a button bank or lister toolbar/menu which
just runs the Compare command. You can drag'n'drop exactly the button you
need from the example button bank which comes with compare.module in the
archive (the Installer script will have asked you if you want it or not
and given you further instructions).
```

1.7   compare.module: arguments

```
Arguments:
-------------------------------- -- -  -
The Compare command installed by the module only has the following
arguments:

BUFFSIZE (OPTIONAL KEYWORD)
--------
This keyword allows you to specify the size, in bytes, of each of the two
buffers used during comparison. If you do not give the argument a default
size of 30k will be used (i.e. 60k in total).

For example,  compare BUFFSIZE 512000  to use two 1/2 meg buffers.

NOSIZEWARN (OPTIONAL SWITCH)
----------
If you give this switch you will not be shown a warning requester before
comparing two files which are not the same size.

For example,  compare nosizewarn  or  compare buffsize 10240 nosizewarn .
```

1.8   compare.module: usage

```
                Usage:
-------------------------------- -- -  -
To do its work, compare.module needs to know which files you want to
compare. When you execute your function which runs the Compare command
the module will:-

o compare the first two files which are selected in the source lister, or,
o if there is only one file selected in the source lister it will compare
  it with the first file selected in the destination lister.

If you haven't selected enough files it will abort.

Note that if you launch the Compare command from a lister's toolbar, that
```

lister is considered the 'source' lister, even if it is not really a
source lister. (This is a good thing, not a bad thing!)

Also note that selected directories in both source and destination listers
are completely ignored.

You can compare two files of different length, but the comparison will
stop at the end of the shortest file. You will be warned about this
before the comparison starts if you choose two such files, unless you use
the
                    NOSIZEWARN argument
                    .

Progress!

Once it has worked out which files are to be compared, a progress window
will open showing you which files are being compared, how much of them
has been done, and how many differences have been found so far.

The Abort Button

Press the abort button at any stage if you do not want the operation to
continue.

Speed

The comparison algorithm is quite simple and should be reasonably fast,
but if there are a lot of differences in certain patterns the time taken
to perform the comparison can grow quite considerably. Of course, if
there are thousands of differences between the two files you are
comparing, it is unlikely that you'll want to see a full report of them
(see below) and your question of "are they really the same" has already
been answered, so you might as well abort the operation.

Full Reports

Once the comparison has finished you will be told the total number of
differences, and, if there were any, you will have the option of
generating a full report.

When knowing how many differences there are between two files isn't
enough, generating a full report is the answer. Looking at a full report
is similar to looking at a Hex/ASCII dump of a file (e.g. using the Opus
HexRead command) except that there are two files shown in parallel and the
differences between them are colour coded. Also, only those parts of the
file where differences occur are shown -- large portions of identical data
are skipped.

Have a look at the
                    example full report
                    to see for yourself.

When you choose to generate a full report a new progress window will open.
During the comparison the files were split-up into chunks depending on
where the differences were, and the progress bar only gets updated after
each chunk is processed. Full reports can also be several times the size
of the original files. Because of these factors, if there are few

similarities between the two files the progress bar may appear to be
'stuck' for several seconds. Don't worry, it hasn't crashed and it will
continue.

Memory

Also, since the full reports are generated in T: and then read into memory
a second time by the Opus text reader, you may require a very large amount
of memory to create them for very large files or files with a large number
of differences. This behaviour can be changed to some extent if there is
anyone out there who would like it, but I have assumed that most people
won't be interested in seeing all 2 million (literally) differences
between the original Aminet INDEX and an encrypted version or anything
like that, because, frankly, it's not very interesting or useful.

In general, you do not need large amounts of memory when you are just
comparing two files, no matter how big they are -- it's only full reports
which are the beasts. If you run out of memory while a full report is
being generated the operation should abort gracefully and free all memory
which it was using.

1.9 compare.module: history

                 History:
------------------------------- -- - -
v1.0 -- 27/Sep/1996
o Initial release.

v1.1 -- 03/Oct/1996
o Side-by-side

                 full report
                 Hex/ASCII dump of differences implimented.
o Can now compare files of different length.
o Unbusies the destination lister, if any, during the comparison so that
  you are free to use it. (I really hate it when listers are left busy
  when they're not really doing anything as it always seems to happen to
  the lister I was about to use as soon as I had started what I was about
  to do first... grr.)
o Major source code cleanup.
o Catalog strings have changed. (Shouldn't cause any problems because
  there have been no translations yet, but just in case, now you know.)

v1.2 -- 14/Oct/1996
o Removed a potential problem from v1.1 where Opus would crash if you
  closed (or used for certain tasks) the destination lister before the
  comparison was completed. Totally my fault, BTW.
o German language catalog now included & installed by Installer script.
  Huge thanks to Georg Rottlaender for the translations.
o Now ignores selected directories in destination listers.
o You can now

                 specify the buffer size
                 .
o If you change the icon image/snapshot/tooltypes for this AmigaGuide you
  no longer have to worry about the Installer script overwriting it.

```
o I have

                 plans
                 for a special ASCII-compare mode.
o You can now disable the warning requester shown before you compare two
  files of different size by giving the
                 NOSIZEWARN argument
           .
```

1.10   compare.module: the future

```
The Future of compare:
------------------------------- -- -  -
At some point I plan to add a 'diff'-style algorithm for comparing ASCII
files which will be able to tell you which lines need adding, removing,
and changing.

Adaptive buffer-size code has also been thought of, although I don't have
a clue how to do it and I'm not sure how much of a difference it would
make.

I don't really have more plans for new features, but if there is something
you don't like or something you would like to see added, please let
me know and I'll consider adding it. Optionally writting full reports
to a temp directory on the harddrive instead of to T: is something I'd
be happy to add if anyone out there wants it.

More modules?
------------------------------- -- -  -
As well as many other add-ons for Opus including numerous ARexx scripts,
the wonderfully simply yet extremely useful (IMO) CDOpus (in Assembler),
the very-nice-indeedy (IMO) hotlist.module (in C), I have also contributed
to the creation of the MagicWorkbench-DOpus icon set (Trevor Morris
deserves as much if not more credit, although I did initiate the project),
and an AmigaGuide to the ARexx commands of Directory Opus. I also wrote
about half a page of the manual to Opus 5.5 and have myself hidden
somewhere in Opus itself. :-) Yes, I'm Opus obsessed and possessed.
I still manage to have something of a life. Well, you have to eat, don't
you? It sure does beat writting /X BBS utilities, at any rate.

Among other things I want to write the final-be-all-and-end-all xpk.module
for Opus, basically taking my most recent attempt (DOpusPack or whatever
I called it) and making it perfect, which I can finally do now that the
OpusSDK is here. I'll probably include xfddecrunch.library support too,
for the hell of it. Finding the time is going to be a problem though. :-(

After that I may disolve a few small ARexx scripts (and DOpusFuncs) into
a 'C' module, and then maybe write a beavis.module who follows the
mouse pointer with his eyes, speaks randomly, and moves his mouth when
he speaks.
```

1.11   compare.module: source code & updates

```
Source code & Updates
------------------------------- -- -  -
The source code (SAS/C) should be available in a separate archive from
the same place you found the main archive in which this guide came from.

You should be able to find it on the GPSoftware web page in the Opus 5
section under Support.

http://www.livewire.com.au/gpsoft/index.html


You should also be able to find it on Aminet in biz/dopus.


...and it's likely that my Web Page will also point to the files in
my own Opus 5 section:

http://users.ox.ac.uk/~kebl0364


Updates of this and other Opus-related things done by myself should
turn up in the same places, although MWB_DOpus will be in pix/mwb on
Aminet, not biz/dopus.
```

1.12 compare.module: credits

```
               Credits
------------------------------- -- -  -
Written by Leo 'Nudel' Davidson
leo.davidson@keble.oxford.ac.uk
http://users.ox.ac.uk/~kebl0364

Please mail me if you find any bugs, or have any ideas for new features.

Many thanks are due to Nick Christie, Jonathan Potter, and Greg Perry for
their advice, examples, and general help beyond the call of duty.
                                                 Thanks guys!

Directory Opus 5 developed by Jonathan Potter and GPSoftware.
Please support this amazing piece of software and the equally amazing
creators by purchasing it.

Additional ideas, support, and advice
------------------------------- -- -  -
Dave Clarke
Georg Rottlaender (German catalog)
Martin Steigerwald
Ahmed's Kebab Van, Oxford  &  Ali's Kebeb Van, Oxford.
(The 'Babylonian Army of Babshit will kick your arse!)

Recent musical support (in alphabetical)
------------------------------- -- -  -
Bjork      (Debut/Post),
BT         (Ima)   This is amazing!,
```

Corrosion Of Conformity (Wiseblood),
Flotsam & Jetsam   (Drift),
Fugees   (The Score)   "I'll be right out...",
Jamiroquai   (Travelling Without Moving),
Metallica   (Load),
Nine Inch Nails   (Pretty Hate Machine/Quake),
Nirvana   (In Utero/Unplugged),
Rage Against The Machine (Evil Empire),
The Prodigy   (Experience/Music For The Gilted Generation),
Tool   (Opiate/Undertow).

This guide was tested with
                CheckGuide
                1.0 by Eddy Carroll
No "AmigaGuide editors" were used. IMO they're as much use as HTML editors.

My setup:
------------------------------- -- - -
 A500, 68000, 1/4 meg chip/fast, 300mb of harddrives, OS3.1
A1200, 68030 50mhz FPU MMU, 2/8 meg chip/fast, one 810mb harddrive, OS3.0
 (And a bloody nice Wizard Developments 3-button mouse sitting on a
  tasteless Hamburger mouse mat!) + an XLink 28.8k Fax/Modem.
(And another A500 which doesn't work very well anymore...)

Ego-Ramblings...
------------------------------- -- - -
I run DOpus5 in workbench replacement mode (please not you don't HAVE to)
on a 64 colour workbench using the dark MWB palette and Pal: Hires,
720 * 282 (overscan). One day I'll have a seventy-eight inch monitor
which can sync down to 1hz. One day, just after I take over the world.
My current backdrop picture is of Babylon 5 :-)

MCP does almost all of my system enhancing -- it rulez.

I'm a registered MUI user and I think anyone who doesn't like MUI needs
their head examining, or maybe they should stop using a 68000-based
machine. Not that I really want to start this war yet again, but even
if you ignore how amazing you can make a MUI look (which is a
quarter of the reason I like it), I have yet to see any other GUI
engine produce anything which impressed me, and I doubt that a GUI
written with anything else which was as complex as the average MUI
and which worked as well would take less time to load. As for
redraw/refresh times, I don't know what the anti-MUI people are
talking about, and, to be honest, I don't think they do either. :-)
(Sorry, I just couldn't resist it.) That isn't to say I think
everything should be written with MUI -- for some tasks it is
overkill, IMO. There are also several extremely good programs which use
'other evil GUI engines', although I'd rather they used MUI :-).

I'm also a registered Thor user and I like Thor a lot, although I do
recognise its faults. It's certainly not the perfect emailer, but it's
extremely good at the same time, and it's my choice.

I still use CygnusEd for editing. It's getting old now and I'd like to
use something really flash like GoldEd, but only when and if it supports
REAL TABS (dream, dream) and irons out a couple of really quirky things
about how it works. I have to respect GoldEd and its author, even if I

don't use the program because I cannot live without some key features.
CEd still does the job, despite looking like an MSDOS program.

My C programming is done in SAS/C which I recently got at the ridiculously
low price which the remaining copies are being sold-off at right now.
I have special buttons setup on my lister toolbar for compiling/making,
running Scopts, cleaning source directories up, running Cpr, and
jumping CEd to errors via a nice ARexx script written by my friend
Nick Christie, someone I owe many pints for all the help getting me
started with C which meant I could get a job this summer and make some
money.

Other things I run of note are PowerSnap, CycleToMenu (don't like MCP's),
ScreenWizard (so cool, yet no-one seems to know about it), KingCON,
TinyMeter, RexxEvent, GrabIFF, MagicMenu, MagicLayers (which is great!),
Beavis!, and BlueEyes (so cute).

My assigns are all done with Assign prefs (not MCP), and I use ILock
to keep people out of my system from time to time. WaitForValidate
stops my harddrive going crazy sometimes, and Super-Select and PicBoot
keep me entertained with random bootpictures from my collection of
around 550 pictures.

I also have a stupid amount of samples, most of which are just wasting
space as I haven't thought of anything to do with them (I can't stand
random boot samples!). The Beavis'n'Butthead samples do get a lot of
use, though. :)

Well, I hope you found all that Really Fascinating. And now for
something completely different...

   As the economic situations and standards of living of millions around
the World fall, people once more begin to blame those old, easy targets.
They contradict: so proud of their "great" country, but still blaming
minorities for how bad things are. WAKE UP. Your country is shit because
of the system; because of what has been happening for hundreds of years;
because of those who are easily led into red-herring causes while the real
problem is allowed to continue, unnoticed by the masses.
   More people don't screw up a country, mismanagement does. Britain
is mismanaged, Germany is mismanaged, America is mismanaged. The entire
World is mismanaged. And it isn't because of the Jews, the Blacks, the
Asians, or the Refugees, it's because while fools blame the innocent
scapegoats the real criminals continue.
   Any educated human being with half a brain can logically work out
that the ideas of Nazis can solve nothing and only set back the progress of

```
humanity.                      __a$\mathrm{\mu}$æææÑÑÑÑææm$\mathrm{\mu}$___
                   __æÆØØØØØØØØØØØØØØØØØØØØÑm__
             aæØØØØØ@\textdegree{}"¯ __      ¯\ensuremath{\lnot}"¶M#ØØØØm_
          _æØØØØ@"       _ÆØØm_          \ensuremath{\lnot}\textdegree{}ÑØØØÑw_
        $\mathrm{\mu}$ØØØ#\textdegree{}      _æØØØØØØp            \ ↩
            ensuremath{\lnot}¶#ØØÑw
      ÆØØØ"        _ÆØØØØØØ#"          _æ#_    ¶ØØØm
    _ÆØØ@        ,dØØØØØØ#"           _ÆØØØP_æØØØØØØØÑ_
    ØØØF          ¶ØØØØØ#_         _ÆØÑ\textdegree{}aæØØØØØØØ#ØØ&
   ÆØØ@            \textdegree{}#ØØØØ#w   _ø@5$\mathrm{\mu}$ÆØØØØØØØÑ¶_ \ ↩
        ensuremath{\lnot}#ØØK
```

```
        JØØØ              ¶ØØØØØØÑw¤Ç$\mathrm{\mu}$ÆØØØØØØ@5æØ#w \ensuremath{\ ↩
           lnot}ØØØ
        aØØ#      _          \ensuremath{\lnot}¶Ø#¶5æØØØØØØØP"  ¶ØØØØØÑ #ØØL
        ØØØF  _aÆØm_        _$\mathrm{\mu}$æØØØØØØÑ\textdegree{}¯        \ ↩
           ensuremath{\lnot}ÑØØP   0ØØQ
        ØØØL  #ØØØØÑ_   _$\mathrm{\mu}$ÆØØØØØØ@5æÆÑ_           \textdegree{}      ↩
           ÆØØ#
        ÆØØE   ¶ØØØ#M_$\mathrm{\mu}$ØØØØØØØP" ÆØØØØØm_            ØØØF
        JØØ#    \ensuremath{\lnot}®$\mathrm{\mu}$æØØØØØØÑ¶®æ"  \ensuremath{\ ↩
           lnot}#ØØØØØm_        JØØØÞ
         0ØØK _$\mathrm{\mu}$ÆØØØØØØ@®$\mathrm{\mu}$Æ#"        \textdegree{} ↩
             ØØØØØØm        ÆØØ#
        \ensuremath{\lnot}#ØØÆØØØØØØØM5æØØ#"        $\mathrm{\mu}$ØØØØØØØ" ↩
              æØØØ
         \ensuremath{\lnot}ØØØØØÑ\textdegree{}¯  \textdegree{}##"         $\ ↩
            mathrm{\mu}$ÆØØØØØØP        dØØØ`
         \ensuremath{\lnot}#ØØb_     \ensuremath{\lnot}          _ØØØØØØØP ↩
              _ØØØ#
           ¶ØØØm_            aÆØØØØØØK        _ÆØØØP
           \ensuremath{\lnot}ÑØØØæ$\mathrm{\mu}$$_        \ensuremath{\lnot}# ↩
             ØØØØP¯     _$\mathrm{\mu}$ÆØØØ@`
            \ensuremath{\lnot}¶#ØØØÑm___     \textdegree{}#\textdegree{}    ↩
             __$\mathrm{\mu}$æØØØØØP"
             "¶ÑØØØØØØMÑæææÑÆØØØØØØØP"
                "\textdegree{}¶#ÑØØØØØØØØ#@\textdegree{}"   You know ↩
                 it makes sense.
```

1.13 CheckGuide by Eddy Carroll

        CheckGuide 1.0 -- Checks AmigaGuide files for common errors

         by Eddy Carroll, September 1994. Freely distributable.

Introduction

If you've ever tried creating even a modestly sized AmigaGuide file by hand,
you'll have discovered that it can be quite difficult to keep track of all the
various links and cross references. As your help file gets larger, it gets
increasingly harder to ensure that you have remembered to reference all the
nodes in the file from at least one place, and haven't left any unresolved
links.

CheckGuide is a simple utility that can detect several common errors in
AmigaGuide files. The errors it can detect are:

· Extra plain text between an ENDNODE and NODE (AmigaGuide won't allow such
  text to ever be displayed)

· Nodes that are defined but not referenced from any other node. Unless your
  program directly instructs AmigaGuide to access such nodes, the user won't
  easily be able to access them.

· Links to help nodes that aren't defined within the file.

· Open braces with no preceding @ character (usually a simple typo).

· @{" Button "} constructs with no LINK command (an easy mistake to make if
  you're entering a lot of them).

All of these options can be selectively enabled or disabled.

          CheckGuide was used in the creation of THIS document!

1.14 Purchasing Directory Opus 5.5

                    Purchasing Directory Opus 5.5
------------------------------ -- - -
Directory Opus 5.5 can be orderred direct from GPSoftware, or from one of
the distributors listed below.

If you already own a previous version of Directory Opus you may be able
to upgrade to version 5.5 for a reduced fee.

For further information about Directory Opus 5.5 and purchasing, please
mail Greg Perry <zzgperry@mailbox.uq.oz.au> or check the GPSoftware Web
Page <http://www.livewire.com.au/gpsoft/>.

Please note that France-Festival-Distribution (FFD) no longer distribute
any GPSoftware products. Read this
                    press release
                    for further details.

 Dr Greg Perry   Ph/fax: +61 7 3661402
 GPSoftware
 PO Box 570
 Ashgrove
 Qld
 Australia 4060

 Small-Biz Software   Phone: +61 (74) 919 190
 PO Box 24   Fax: +61 (74) 926 860
 Golden Beach
 Queensland
 AUSTRALIA 4551

 Wizard Developments   Phone: +44 (0)1322 527 800
 PO BOX 490   Fax: +44 (0)1322 527 810
 Dartford
 Kent DA1 2UH
 ENGLAND

 Schatztruhe   Phone: +49 201 788 778
 Veronikastr 33   Fax: +49 201 798 447
 45131 Essen
 GERMANY

```
 Micro R&D      Phone:  +1 308 745 1243
 721 'O' Street    Fax:  +1 308 745 1246
 Loup City
 NE 68853
 USA
```

1.15 Official Notice From GPSoftware: FFD

```
GPSoftware French Distribution - France-Festival-Distribution
------------------------------ -- -  -
After recent actions by FFD, we have officially withdrawn all of our
products and licences from FFD and refuse to let them ever be associated
with any of our products again. I have advised all of my other distributors
that they may not supply FFD with any of our products. Further, being the
copyright holder of trademarks, product names and other copyrights for
our products, we officially withdraw permission for FFD or any associated
company to use the name Directory Opus, GPFax, or GPSoftware in any
advertising or related activities.

Because of unreconciled differences, FFD is no longer a distributor or
reseller of GPSoftware products and GPSoftware will no longer honour
warranty or support for any of our products purchased from FDD after
30th June 1996.

Dr Greg Perry, GPSoftware 11th September 1996
```