Default

Adam Wasiak

Copyright © CopyRightÂ©1996 Adam Wasiak - GhostSoft

COLLABORATORS

	TITLE : Default	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Adam Wasiak	July 20, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1 Default **1**

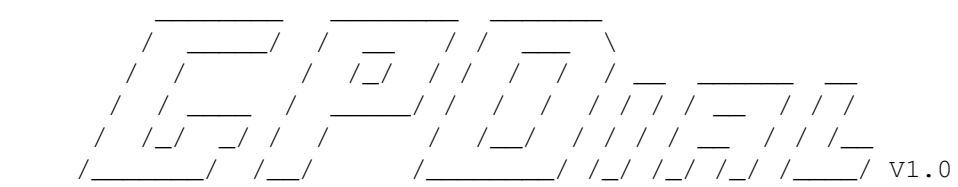
1.1 GPDial v1.0 . . . 1
1.2 GPDial v1.0 - Disclaimer . . . 1
1.3 GPDial v1.0 - Requirements . . . 2
1.4 GPDial v1.0 - What is it . . . 2
1.5 GPDial v1.0 - Installation . . . 3
1.6 GPDial v1.0 - Using GPDial . . . 4
1.7 GPDial v1.0 - User Interface . . . 5
1.8 GPDial v1.0 - Scripts . . . 5
1.9 GPDial v1.0 - Scripts - Accept . . . 6
1.10 GPDial v1.0 - Scripts - Close . . . 6
1.11 GPDial v1.0 - Scripts - End . . . 6
1.12 GPDial v1.0 - Scripts - Execute . . . 6
1.13 GPDial v1.0 - Scripts - Extract . . . 7
1.14 GPDial v1.0 - Scripts - Goto . . . 7
1.15 GPDial v1.0 - Scripts - Hide . . . 7
1.16 GPDial v1.0 - Scripts - if/else/endif . . . 7
1.17 GPDial v1.0 - Scripts - Lab . . . 8
1.18 GPDial v1.0 - Scripts - Math . . . 8
1.19 GPDial v1.0 - Scripts - Pause . . . 8
1.20 GPDial v1.0 - Scripts - Print . . . 8
1.21 GPDial v1.0 - Scripts - Printerror . . . 9
1.22 GPDial v1.0 - Scripts - Quit . . . 9
1.23 GPDial v1.0 - Scripts - Remchar . . . 9
1.24 GPDial v1.0 - Scripts - Remove . . . 9
1.25 GPDial v1.0 - Scripts - Send . . . 10
1.26 GPDial v1.0 - Scripts - Set . . . 10
1.27 GPDial v1.0 - Scripts - Setser . . . 10
1.28 GPDial v1.0 - Scripts - Show . . . 11
1.29 GPDial v1.0 - Scripts - Wait . . . 11

1.30 GPDial v1.0 - Variables 12
1.31 GPDial v1.0 - Script Errors 13
1.32 GPDial v1.0 - Registration 15
1.33 GPDial v1.0 - Things you should know about 16
1.34 GPDial v1.0 - Contacting me 17
1.35 GPDial v1.0 - Readme.First 18

Chapter 1

Default

1.1 GPDial v1.0



Disclaimer - It has to be said I'm afraid
Requirements - What should GPDial run on
What is it - So what the bloody hell is GPDial
Installation - How to install GPDial
Using GPDial - Basic usage
User Interface - Its GUI
Scripts - How to write a GPDial script
Variables - Variables that GPDial understands
Script errors - Error messages from scripts
Registration - Register for improved capabilities
Things to watch for - Things to remember
Contacting me - How to get hold of me

1.2 GPDial v1.0 - Disclaimer

Disclaimer

This software is provided as-is, without warranty of any kind, either expressed or implied. In no event will the author be liable for direct, indirect, incidental or consequential damages or data loss resulting from the use or application of this software. The entire risk as to the results and performance of this software is assumed by the user.

This software has been made available free of charge. It is not public domain, shareware, beerware or any other ware anyone wishes to make up.

This software may be distributed freely as long as it remains in its original state, which is:

```
  Archive name - GPDialv1.0.lha

  Contents of archive

  Install.GPDial
  Install.GPDial.info
  Replace
  Replace.doc
  Readme.First   <----------- You should have already read this file.
  Readme.First.info
  bin/GPDial                   If not read it now.
  bin/GPDial.info
  doc/GPDial.guide
  doc/GPDial.guide.info
  doc/main.iff
  doc/entry.iff
  doc/config.iff
  doc/progress.iff
  Scripts/startup.GPDScript
  Scripts/Default.GPDSript
  Scripts/Disconnect.GPDScript
  Files/GPDial.config
  Files/GPDial.book

  Despite GPDial v1.0 being free of charge it is still © 1996 Adam Wasiak
```

1.3 GPDial v1.0 - Requirements

```
Requirements

  GPDial requires OS 2.04 or above. It will normally be used in
  conjunction with AmiTCP.

  I developed GPDial using

      o CygnusEd Pro V3.5
      o Lattice C v5.01
      o GadToolsBox v2.0b
      o DPaint 4 AGA
      o BrushCon v1.20

  It was tested on an A1200 running

      o WB 3.0
      o AmiTCP v2.2 & 2.3
      o rhslip.device

  Note: Although GPDial should run on OS 2.04 it has not been tested on
        such a machine. It has also not been tested with a ppp.device.
```

1.4 GPDial v1.0 - What is it

So What Is It?

Well after several hours of installing AmiTCP and finally getting it to work I set about looking for a dialer. There are several dialers out there, but none of them met my needs. So GPDial was born.

GPDial is a script based dialer that has a fancy GUI. It has the capability of storing several entries in its phone book. So if you have more than one service provider, it could be the dialer for you.

GPDial doesn't necessarily have to be used with AmiTCP. It can be used to automate any sequence of events. Especially ones involving the serial port.

1.5 GPDial v1.0 - Installation

Installation

The installation is quite straight forward. There is an installation script that will do everything for you. To run it type

```
installer Install.GPDial
```

or double click on its icon.

The installation script will copy GPDial to a directory of your choice the default is AmiTCP:bin

Replace will then be copied to a directory of your choice the default is C:. Replace is a find and replace utility. See its doc for more info.

The config files are then copied. GPDial will look for the config file in the following places:

```
The location held in environment variable GPD_CONFIG
AmiTCP:db/
INet:db/
S:
```

The installation script will generate the GPD_CONFIG environment variable for you in ENV: and ENVARC:

The next step involves copying 2 GPDial scripts into a directory of your choice.

Finally the doc files are copied to a place of your choice. The GPDial doc (this file) has 4 iff pictures which will also be copied, it might be best if the guide and the iffs are placed in a directory of their own (it would certainly be neater). The Replace doc is also copied to a directory of your choice.

and thats it.

1.6 GPDial v1.0 - Using GPDial

Using GPDial

GPDial can be run from either the cli or workbench. It must be noted however that GPDial requires a stack of 10000 bytes. This is handled by its icon if run from workbench. If you run it from the cli remember and change the stack with the command:

```
stack 10000
```

I use GPDial from a menu using "tools daemon". This supports different stack sizes.

If GPDial is run from the workbench a window will open.
see the User Interface section for more details.

GPdial has a few command line arguments. They are:

```
GPDial [-options] [script | phonebook entry]

      options
         n         No startup script
         c         Don't open serial port on startup
         e         Run script associated with the phonebook entry

       script      Name of a script to run
```

Normally no arguments are required.

If a script is given then the user interface is not displayed.

GPDial normally opens the serial port as soon as it starts. This can be avoided by using the 'c' flag. (the serial port can be opened latter on with the SETSER script command).

After GPDial has allocated all resources it will attempt to run the startup script (if there is one). To stop this happening use the 'n' flag.

If the 'e' flag is given then the script associated with the phonebook entry is executed. There are several benefits of doing this.

When a script is executed from the User Interface several special variables are filled in. See GPDial Variables section for more details. If a script is executed from the command line a few of these variables a not filled in. This is overcome by using the 'e' flag. It pretends the script has been executed from the interface.

It should be noted that when using the 'e' flag you also enter the phonebook entry name to execute, NOT a script name.

1.7 GPDial v1.0 - User Interface

```
The User Interface

  When GPDial starts the Main window is displayed.

  The Config window is used to edit certain details and the name
  of the startup script.

  To add or edit a Phonebook entry the Phonebook edit window is used.

  When a script runs the SHOW command will display the Progress window.

  The close gadget on the progress window will not terminate the script.
  It does exactly the same as the HIDE command.

  If the abort gadget is clicked the script will terminate. If however
  the gadget is clicked while the modem is dialing or connected the
  connection will continue to exist. Abort merely terminates the script.
```

1.8 GPDial v1.0 - Scripts

```
GPDial Scripts

Here is a list of all the scripting commands

     ACCEPT           - Get input from the user
     CLOSE            - Close the serial port
     END              - End of the script
     EXECUTE          - Execute a command
     EXTRACT          - Extract a word from a string
     GOTO             - Goto a Label
     HIDE             - Hide the progress window
     IF               - Selection
     MATH             - Do a calculation
     PAUSE            - Pause for a length of time
     PRINT            - Print a message to the progress window
     PRINTERROR       - Display an error message
     QUIT             - Quit the script
     REMCHAR          - Remove a character from a string
     REMOVE           - Remove the progress window
     SEND             - Send a string to the serial port
     SET              - Set a variable
     SETSER           - Change the serial port settings
     SHOW             - Show the progress window
     WAIT             - Wait for a string from the serial port

  Special Control Characters
  Several characters can not be displayed normally. In order to display
  them or send them to the serial port you should use the control
  character sequence. They are as follows:

    \n - Carriage return
    \r - Newline
```

```
    \f - Formfeed
    \" - The quote char "
    \$ - The dollar char $
    \ - The slash char \

  An extra control char is also available to the SEND command:

    \w - wait for 1 second.

  Remember to end all commands to the modem with a \r.
```

1.9 GPDial v1.0 - Scripts - Accept

```
ACCEPT <string> [HIDE] [BLANK] [label]

  This command will prompt the user for input.

  The result is placed in a variable called GPD_INPUT.

  If the "HIDE" option is given then the text entered will be hidden
  (ideal for passwords). The label will be jumped to if "Cancel" is
  clicked, if it is omitted then the script will continue as if "Ok"
  was clicked.

  Not entering a string in the requester will simulate clicking
  "Cancel" unless the "BLANK" option is activated.
```

1.10 GPDial v1.0 - Scripts - Close

```
CLOSE

  This command will force the release of the serial port.
```

1.11 GPDial v1.0 - Scripts - End

```
END

  The end of a script.
```

1.12 GPDial v1.0 - Scripts - Execute

```
EXECUTE <command> [label]

  Execute an AmigaDOS command, if the command fails the label will be
  jumped to.

  It should be noted that only system errors will cause control to
```

```
  jump to the label. If actual command returns an error code
  it will be ignored.

  Any other output from the command will also be ignored.
```

1.13 GPDial v1.0 - Scripts - Extract

```
EXTRACT <string> <position> [label]

  This command will take the word at <position> in the <string> and
  place it in a variable called GPD_VALUE.

  The label will be jumped to if there are not enough words in the string
  for the operation.
  e.g.

  EXTRACT "your IP address is 146.176.164.13" 5
  PRINT $GPD_VALUE

  gives:

  146.176.164.13
```

1.14 GPDial v1.0 - Scripts - Goto

```
GOTO <label>

  Jump to a label. Labels are denoted with the LAB command.
```

1.15 GPDial v1.0 - Scripts - Hide

```
HIDE

  This command will hide the progress window.

  It can be redisplayed with a SHOW command.
```

1.16 GPDial v1.0 - Scripts - if/else/endif

```
IF <operand1 operator operand2>
..
ELSE
..
ENDIF

  Checks to see if the two operands match.
```

```
  Valid operators are:

  =   Check for equality        - Strings and Numerics.
  >   Greater than              - Numerics only.
  <   Less than                 - Numerics only.
  >=  Greater than or equal to  - Numerics only.
  <=  Less than or equal to     - Numerics only.
  !=  not equal to              - Strings and Numerics.
```

1.17 GPDial v1.0 - Scripts - Lab

```
LAB <label:>

  This command is normally just skipped over. Other commands
  use it to allow control to jump to different parts of the script.
```

1.18 GPDial v1.0 - Scripts - Math

```
MATH <variable> <=> <operand1> <operation> <operand2>

  This command allows you to do simple maths.

  Operations are:
  +   Plus
  -   Subtract
  *   Multiply
  \   Divide

  e.g.
  SET SUM TO 1
  MATH SUM = $SUM + 1
  PRINT $SUM

  gives:
  2
```

1.19 GPDial v1.0 - Scripts - Pause

```
PAUSE <time>

  This command will pause the script for the stated number of seconds.
  The <time> must be numeric and greater than 0.

  A PAUSE length of less than or equal to 0 is invalid.
```

1.20 GPDial v1.0 - Scripts - Print

```
PRINT <string>

  This command displays a string in the progress window.
```

1.21 GPDial v1.0 - Scripts - Printerror

```
PRINTERROR <string>

  This command display the string as an error message.
```

1.22 GPDial v1.0 - Scripts - Quit

```
QUIT

  This command will quit GPDial.
  If the script was run from the GUI it will also be ended.

  This command could be used on successful connection to your service
  provider.
```

1.23 GPDial v1.0 - Scripts - Remchar

```
REMCHAR <string> <value>

  This command will remove the char at the given position and will
  shuffle the rest of the chars down.
  If you want to remove the last char of the string and you don't know
  its length you can use a value of -1.

  The resultant string is placed in the variable GPD_TEXT

  e.g.

  SET test to "hello there"
  REMCHAR $test 6
  PRINT $test

  gives:

  hellothere
```

1.24 GPDial v1.0 - Scripts - Remove

```
REMOVE [time]

  This command removes (closes) the progress window after "time" seconds has
  expired. It is removed immediately if the time is not stated.
```

```
  The time value must be greater than 0.
```

1.25 GPDial v1.0 - Scripts - Send

```
SEND <string> [HIDE]

  This command will sends a string to the modem.
  A message is displayed in the progress window everytime a SEND is
  executed. The HIDE flag can be used to stop this. It is ideal for
  passwords.
```

1.26 GPDial v1.0 - Scripts - Set

```
SET <variable> <TO> <string>

  This command will set a variable to a string. Numeric variables are
  also treated in this way

  e.g.

  SET text TO "Hello"
  SET value1 TO "1"
  SET result TO "0"
  MATH result = $value1 + "3"
  PRINT $text
  PRINT $result

  gives:

  Hello
  4
```

1.27 GPDial v1.0 - Scripts - Setser

```
SETSER <baud> <data> <stop> <parity>

  This command will open the serial port if it is closed.
  It will also change the serial port settings.

  baud   -      300
                600
               1200
               2400
               4800
               9600
              19200
              38400
              57600
              76800
```

```
            115200

  data   - 7    7 data bits.
           8    8 data bits.

  stop   - 1    1 stop bit.
           2    2 stop bits.

  Parity - N    No parity.
           O    Odd parity
           E    Even parity.
```

1.28 GPDial v1.0 - Scripts - Show

```
SHOW [ABORT] [ALL]

  This command shows the progress of the connection in a progress window,
  the options are:

     ABORT - This gives an abort button to abort the script. Care should
             be taken with the abort feature. If it is clicked while the
             modem is dialing or connected the connection will continue
             to exist. Abort merely terminates the script.

     ALL   - This shows ALL output. Normally only PRINT statements and
             the odd system message will be displayed in the progress
             window. If ALL is stated then characters received from the
             modem during a WAIT or WAITLINE are also displayed.
```

1.29 GPDial v1.0 - Scripts - Wait

```
WAIT <"string"> [OR <"string"> ..] [timeout <label>]   (max of 4 strings)

  This command will waits for a string to be received from the serial port.
  The string received is placed a variable called GPD_STRING, and the
  whole line with the string in is placed in a variable called GPD_LINE.

  There is a maximum of 4 strings that can be waited for. They are
  separated with the "OR" statement.

  An optional timeout is available. If this time limit runs out before
  the string is received control jumps to the label.

  WAIT will wait for a string to be received and then continue with the
  script. If however you wish to wait for a whole line (256 chars or
  newline terminated) then WAITLINE should be used in place of WAIT.

  e.g.

  WAIT "CONNECT" OR "BUSY" OR "ENGAGED" 30 Error

  or
```

```
  WAITLINE "your IP address is" 10 Error
```

1.30 GPDial v1.0 - Variables

```
GPDial Variables

This is a list of variables that GPDial knows about. More variables can be
added by scripts. These are deleted automatically when the script ends.

Variable name   Description                          Default value
GPD_BAUD        Holds the current Baud rate

GPD_DATA        Holds the current number of data
                bits

GPD_STOP        Holds the current number of stop
                bits

GPD_PARITY      Holds the current parity used

GPD_INIT        Holds the initialisation string      "AT\r"

GPD_DTROFF      Holds the string to ignore DTR      "AT&D0\r"

GPD_BUSY        Holds the busy string                "BUSY"

GPD_CONNECTED   Holds the connected string           "CONNECT"

GPD_HANGUP      Holds the hangup string              "\w\w+++\w\wATH\r"

GPD_PREDIAL     Holds the dial prefix                "ATDT"

GPD_SUFDIAL     Holds the dial sufix                 "\r"

GPD_PHONENUMBER Holds the number of the site that
                is being called.

GPD_SITE        Holds the name of the site that
                is being called. If you don't
                run the script from the GUI than
                this variable will be blank.

GPD_LINE        This variable is filled after every
                WAIT or WAITLINE command succeeds.
                It holds the whole line that the
                command succeeded on.

GPD_USERNAME    This is the username for the system
                you are calling. It is filled in
                from the phonebook. If you don't
                run the script from the GUI than
                this variable will be blank.

GPD_PASSWORD    This is the password for the system
```

```
                you are calling. It is filled in
                from the phonebook. If you don't
                run the script from the GUI than
                this variable will be blank.

GPD_STRING      This variable is filled after every
                WAIT or WAITLINE command succeeds.
                It holds the actual string that the
                command succeeded on.

GPD_VALUE       This variable is filled after every
                EXTRACT command succeeds

GPD_TEXT        This variable is filled in by
                the REMCHAR command.

GPD_INPUT       This variable will be filled with
                the results of an ACCEPT.

GPD_AUTHOR      The name of the author of GPDial    "Adam Wasiak"

GPD_NAME        The name of the program             "GPDial"

GPD_VER         The version information for GPDial  "V1.0 ©1996 GhostSoft"

GPD_REG         Holds the registration details      "<unregistered>"
```

1.31 GPDial v1.0 - Script Errors

```
Script Errors

  This is a list of error messages that can be produced by incorrect
  scripts.

Invalid command
  You have tried to use a command that GPDial does not understand.
  It is probably just a spelling mistake (I make them all the time).

Missing LAB:
  You tried to do a GOTO to a label that does not exist. This error
  can be produced by GOTO ACCEPT WAIT or WAITLINE.

PAUSE length incorrect
  You entered a PAUSE length that wasn't numeric or was 0.

REMCHAR statement incorrect
  The value part of the REMCHAR command is incorrect. It must be
  either -1 or greater than 0.

Invalid operator
  The operator part of the IF statement is incorrect.
  Valid operators are: = != < > <= >=

Out of memory for SET statement
  Your machine ran out of memory while trying to SET a variable.
```

```
SET statement incorrect
  One of the SET statements is incorrect.

MATH statement incorrect
  One of the MATH statements is incorrect.

EXECUTE statement incorrect
  One of the EXECUTE statements is incorrect.

EXTRACT statement incorrect
  One of the EXTRACT statements is incorrect.

GOTO statement incorrect
  One of the GOTO statements is incorrect.

IF statement incorrect
  One of the IF statements is incorrect.

LAB statement incorrect
  One of the LAB statements is incorrect.

PRINT statement incorrect
  One of the PRINT statements is incorrect.

PRINTERROR statement incorrect
  One of the PRINTERROR statements is incorrect.

SEND statement incorrect
  One of the SEND statements is incorrect.

MATH attempted a divide by 0
  A MATH statement attempted to divide by 0

MATH variable not declared
  Attempted to assign a value to an undeclared variable in a MATH
  statement.

ACCEPT must have a string
  An ACCEPT statement was encounter that did not have a <string>.

Timeout length incorrect
  WAIT or WAITLINE timeout length that wasn't numeric or was 0.

<baud> in SETSER is incorrect
  The baud rate part of the SETSER command should be:
       300
       600
      1200
      2400
      4800
      9600
     19200
     38400
     57600
     76800
    115200
```

```
<data> in SETSER is incorrect
  The data part of the SETSER command should be
      7    7 data bits.
      8    8 data bits.

<stop> in SETSER is incorrect
  The stop part of the SETSER command should be
      1    1 stop bit.
      2    2 stop bits.

<parity> in SETSER is incorrect
  The parity part of the SETSER command should be
      N    No parity.
      O    Odd parity.
      E    Even parity.

Serial device closed
  This error occurs when you try to perform an operation on the
  serial device when it is closed. It is produced by
      SEND
      WAIT
      WAITLINE

  NOTE: SETSER will reopen the serial device if it is closed.

Can't write to serial port
  A SEND command failed for some unknown reason.

Can't reopen serial device
  One of the SETSER statements has failed to reopen the closed serial port.

SETSER can't change settings
  A SETSER command failed to change the serial port settings.

Can't open progress window
  One of the SHOW failed to open the progress window.

Too many "OR"s
  One of the WAIT or WAITLINE statements has too many ORs.
```

1.32 GPDial v1.0 - Registration

```
Registration

  This version of GPDial is absolutely free. See disclaimer for more info.

  I am currently working on a new version of GPDial. This will be
  registration only and available early in April 1996.

  The new version will have the following features:

  o Multiple phone numbers per site. (for sites with multiple lines
                                      each with different numbers).
  o Arexx port.
```

```
  o More scripting commands.
      o Logfiles.
      o Loops.
      o Ask (YES/NO) requester.
  o Variable progress window size.
  o Enhanced abort processing.
  o Option to use a device other than the serial.device.
  o Hotkey which will bring up the progress window.
  o Encrypted and hidden passwords.
  o Password protection.
  o Any other ideas that are mailed to me.

  To register you should send 10 pounds sterling to me.
  If you don't have EMail you should also send a S.A.E. big enough to
  hold a disk. In return I will send you a key file. You will also need
  to download GPDial1.1_reg.lha.

  To register all I will need to know is your name and address. If
  you have an EMail address you should also send that too. If you are
  registering on behalf of a company you should also send the company
  name.

  NOTE: v1.1 is still under development and will be released in early
        April 1996. Any registration fees I receive before then will be
        held untill it is released.
```

1.33 GPDial v1.0 - Things you should know about

```
Things You Should Know About

  Here are a few things to watch out for.

1.
  Control chars (\n \r \ \" and \$) that occur within the GPD_INPUT
  variable will not be evaluated. This can only happen with the ACCEPT
  command.
  e.g.

  ACCEPT "Enter a string"      ; you entry "Yes\No"
  PRINT $GPD_INPUT

  gives    Yes\No
  and not  Yes\No


2.
  Remember commands that go to the modem must end in \r.
  e.g.

  SEND "AT\r"

3.
  The stack must be greater than 4096 - Recommended value is 10000.

4.
```

When running scripts from the command line it should be noted that
several variables are not filled in. Namely:

GPD_SITE
GPD_PHONENUMBER
GPD_USERNAME
GPD_PASSWORD

If the '-e' option is given the name given is searched for in the
phonebook and the relevant details are filled in.

5.
Scripts will run with a closed serial port until a WAIT or SEND command.

If you attempt to run a script and the serial port is closed (either by
the -c command line arg or the CLOSE command) the serial port settings
for the script (Baud rate, Data bits, Stop bits and Parity) will
not be set. However it can be opened latter on with the following
command:

```
SETSER $GPD_BAUD $GPD_DATA $GPD_STOP $GPD_PARITY
```

5.
During testing I noticed something strange. Five modem settings don't
seem to work. I understand that the serial.device can't handle
certain settings. Here are the settings that don't work:

```
8/1/O
8/1/E
8/2/O
8/2/E
7/1/N
```

I'm not sure if these are the settings that are meant to not work
or if there is a bug in the code. If someone knows please mail me.

1.34 GPDial v1.0 - Contacting me

Contacting Me

Send me any bug reports/queries/questions/comments (good or bad).
I would like to know if you use this program, send me a postcard
or EMail me.

I can be contacted at:

EMail : ghosty@spuddy.mew.co.uk

Snail : Adam Wasiak
10 Chapel Villas
Dalgety Bay
Fife
Scotland
KY11 5UD

1.35 GPDial v1.0 - Readme.First

```
This archive should contain the following executables.

GPDial  - v1.0 of a script based dialer.
Replace - v1.0 of a find and replace utility.

This archive may be freely distributed.

All software in this archive in © 1996 Adam Wasiak - GhostSoft.

It is recommended you read the installation section of GPDial.guide
before you do the install.

NOTE: The GPDial installer will also install the Replace utility for you.
```