Programmer's File Editor Help File

The PFE help file contains full information on the current <u>version</u> of the editor. There is no written documentation, but you will be able to find all that you need both to begin work with PFE, and to use its more advanced features, in here.

When you're using PFE's dialog boxes, you can get help on the exact steps you should take by clicking the **Help** button or pressing the **F1** key.

Like all Windows help files, this one contains many hypertext links which take you from one topic to another in a structured way. Also, most topics set up the **Up** button above the text display area to take you to the most appropriate menu topic, so you can easily retrace your steps from a lengthy session of following links.

If you're not sure how to use the Windows Help System, select the **How To Use Help** item from the **Help menu** at the top of this window, or from PFE's own help menu.

<u>**Introduction To PFE**</u>	This section gives you general information on PFE, tells you where you can get the latest version, and contains notes specific to this release
<u>**Procedures**</u>	This section tells you how to go about doing things in PFE, and explains PFE's features
<u>**Commands**</u>	This section gives you information on each of PFE's menu commands. You can look here for a summary of what a menu command does, then follow the links into the **Procedures** section for a fuller explanation
<u>**Reference**</u>	This section contains reference information: there are lists of the default key bindings, of the various mappable functions, and so on

Introduction To PFE

Welcome to Programmer's File Editor, a programmer-oriented text editor for Windows that has wide application in all areas needing the ability to edit files.

PFE is a standard Windows Multiple Document Interface (MDI) application, which means that you can operate it in the same way as any other standard Windows application. It features many powerful facilities, which you will find described in the sections in this help file.

The program is distributed in two separate editions to suit the system you are running it on:

- The **16-bit Edition** is designed for IBM PC compatible systems running Windows 3.1x or Windows for Workgroups 3.1x

- The **32-bit Edition** is distributed in two separate release sets : one is designed for Windows NT 3.5 on Intel processors and IBM PC compatible systems running Windows 95, and one is designed for Windows NT 3.5 on PowerPC platforms,

The two editions, when run on the systems they are built for, have identical functionality. However, some features will ot be available if, for example, you run the 32-bit Edition under Windows 3.1x with Win32s.

This introductory section gives you an overview of what PFE is, tells you where you can the latest version, and so on. Further information is available in the distribution set:

readme.txt	Basic information to get you started with this release
changes.txt	Details of all the changes made from previous releases
faq.txt	A list of Frequently Asked Questions and the answers to them
problems.txt	Details of any unresolved problems in the current release

What is Programmer's File Editor?
Disclaimer
Where To Get It
Using And Distributing PFE
Notes On This Release
Acknowledgements

What Is Programmer's File Editor?

Programmer's File Editor (PFE) is a large-capacity text file editor, oriented towards those who use Windows as their program development environment, and so incorporates many features that make it a convenient work management system. Although its primarily oriented to program development, it makes a very powerful general editor for any purpose at all.

PFE's capacity is essentially limited only by the total amount of memory available on your system. There are no editor-imposed limits on the number of files that you can edit simultaneously, nor on the number of edit windows that you may have open. There is no limit on the size of file that can be handled, and none on the number of lines that a file may contain.

PFE adheres strictly to the Windows MDI conventions. You can invoke most commands and facilities from menus; you can move around with a mouse or with the standard keyboard shortcuts; you can cut and paste from the clipboard, and so on.

You can reconfigure the use of keys, so that if you don't like the built-in way PFE works, you can change which keys do what to suit your preferences. If you like, you could have commands invoked by two-character key sequences like **Esc G** and **Ctrl+K Ctrl+B**; or you could use key sequences such as **Alt+F** and **Alt+S** - almost all keyboard keys can be mapped, in any combination.

PFE is able to run DOS commands, such as compilers, and to capture their output into windows for inspection. This lets you use it as an integrated development environment, cutting down the amount of work you need to do to build and test your applications. You can also quickly launch the application you're developing, and you can configure details of the Windows tools that you use, so that these too can be launched with only a few mouse clicks.

There are also some features that let you build files more easily. You can define sets of templates - standard lines of text - that you can insert into the file you're editing with just a few mouse clicks. You can group the templates you work with into distinct files, and load them for use automatically.

Why should you use PFE? If you already have a favourite editor, fine. PFE is not sold for money, so the author has no need to persuade you to do something you don't want. However, you might like it; and you are also very welcome to contribute ideas for improvements. There is a great deal of scope for changing and improving things, so let the author know what you think.

Disclaimer

Programmer's File Editor is supplied on an *as-is* basis. The author offers no warranty of its fitness for any purpose whatsoever, and accepts no liability whatsoever for any loss or damage incurred by its use.

PFE is not a supported product. The author accepts no commitment or liability to address any problems that may be encountered in using it; however, PFE is continually being developed and improved, so he is always interested to hear about any bugs or deficiencies.

Where To Get It

Programmer's File Editor is available from a wide range of archive sites throughout the world. New releases are uploaded by the author to three major archives:

- For those at UK Higher Education establishments, copies are placed on the **micros.hensa.ac.uk** archive, which supports Internet ftp, Gopher, World Wide Web and telnet access.

- For others with Internet access, copies are posted to **ftp.cica.indiana.edu** and the Simtel archive held on **oak.oakland.edu**.

The versions on these archives will be the definitive latest releases. Note that files can take a long time to be placed in their final directories on CICA, so check in the uploads area (and any subdirectories this contains) first.

The author also passes releases to the moderator of the **comp.binaries.ms-windows** group on Usenet, and posts announcements to the **comp.os.ms-windows.announce**, **comp.editors** and **comp.os.ms-windows.utilities** groups. (This last group is a good place to discuss PFE with other users, also)

Though not posted there by the author directly, copies will normally be available on **CompuServe** within a few weeks of release, in the **WINSHARE** forum.

From here it will gradually spread to other repositories: if you know of an archive or BBS in your part of the world that doesn't yet have the latest release, please upload a copy there too.

Using And Distributing PFE

Programmer's File Editor may be freely used for any purpose. You may use it privately or in the course of your work; there is no fee, and no registration is required.

You may distribute it to anyone, and you may place it on any archive or bulletin board system. You may not charge anyone for it other than a reasonable fee to cover your distribution costs. Normally, you should distribute PFE in the form as supplied by the author; however, you may repackage it to suit the conventions and needs of an archive or bulletin board system.

PFE may not be distributed as part of any commercial product without a prior licence agreement. Licenses to incorporate the source into commercial products are also available.

PFE may be reviewed for any publication, whether in hard copy or electronic form, without the author's prior permission.

Notes On This Release

Programmer's File Editor is still under active development. Although it is reliable in operation - many people all over the world now use it as their preferred editor, and the author does all the development work on PFE with it - not all the planned facilities have yet been added to it, and some facilities may in the future be provided in a different way.

The author welcomes feedback from users, and ideas for further facilities.

Notes Specific To The 16-bit edition
Notes Specific To The 32-bit edition

Notes Specific To The 16-bit Edition

The 16-bit edition of PFE can be run on Windows 3.1x and Windows for Workgroups 3.1x systems. It is not recommended that you run this version under Windows NT or Windows 95.

You will be able to use <u>key mapping</u> and <u>template files</u> produced by the 16-bit edition with the Intel version of the 32-bit edition, as the formats are identical. However, since normally both systems are installed in the same Windows directory, the file names will differ between the two versions.

The key mapping and template files used by the16-bit edition are not compatible with the PowerPC version of the 32-bit edition in this release.

The format of <u>template files</u> is also common to both editions.

The 16-bit and 32-bit editions run independently of each other. An attempt to start the 32-bit editions will never instead pass control to an instance of the 16-bit edition running under the Win16 subsystem, and vice versa.

Notes Specific To The 32-bit Edition

The 32-bit edition of PFE is released in two formats.

- The Intel version can be run on Windows NT 3.5 on Intel processors, and on Windows 95. It is not recommended that you run this edition on Windows 3.1x systems using Win32s

- The PowerPC version can be run on Windows NT 3.5 on PowerPC systems.

If you are using the Intel version of the 32-bit edition, you will be able to use the same <u>key mapping</u> and <u>template files</u> as the 16-bit edition, as the formats are identical. However, since normally both systems are installed in the same Windows directory, the file names will differ between the two versions.

The key mapping and template files used by the PowerPC version of the 32-bit edition are not compatible with the 16-bit edition or the Intel version of the 32-bit edition in this release.

The 16-bit and 32-bit editions run independently of each other. An attempt to start the 32-bit editions will never instead pass control to an instance of the 16-bit edition running under the Win16 subsystem, and vice versa.

There are currently no plans to produce binary versions of the 32-bit edition for MIPS or DEC Alpha systems.

Acknowledgements

I'm happy to be able gratefully to acknowledge here the help of those who have contributed ideas and testing time towards this project. Bug reports,   comments and suggestions are *always* welcome from *everyone*. I'll try to respond to all e-mail messages; time just may not permit postal replies, I'm afraid, but everything will be read and considered.

Icons and bitmaps were designed by Dave Bleasdale (copyright assigned)

I must express my gratitude and indebtedness to the many people who have sent me suggestions on how PFE could be improved, and details of bugs and problems they've found. I'm especially grateful to the team of beta testers, who have worked with the pre-release versions and helped me iron out the many problems. Thanks, guys.

PFE Menu Commands

The menu bar shown at the top of the main window contains a selection of items that you can use to perform many of PFE's operations. Not all of the menu items on the menu bar, and not all of those on each of the pull down menus, will appear if you don't have a file open - for example, there'll be no **Edit** item on the menu bar if there's no file to edit.

The File Menu	File-related commands that let you open, close and save files, etc.
The Edit Menu	Text-altering commands that let you search for strings and replace them, use the clipboard, etc.
The Options Menu	Commands for setting how windows appear, and for expressing your preferences.
The Template Menu	Commands for handling all aspects of templates
The Execute Menu	Commands for running DOS commands and Windows tools
The Macro Menu	Commands for using keyboard macros
The Window Menu	Commands for arranging, duplicating and selecting windows
The Help Menu	Commands that start the Windows help engine

The File Menu

This item on the menu bar contains file-related options. Not all of the items listed here will appear on the menu if you don't have any files open.

File New
File Open
File View
File Insert
File Save
File Save As
File Save All
File Write
File Name
File Mail
File Close
File Close All
File Print
File Print Setup
File Exit

Most-Recently-Used List

File New

This menu item creates a new, empty edit window in which you can type text.

Default command key: **Ctrl+N**

Tool bar action:

Desktop mouse action: **Ctrl + Click Right Button**

 The window will have no file name <u>associated</u> with it. The tab size and other settings will be the values you set up for "new files" with the **Options Default Modes** command

About Creating a New File

File Open

Opens one or more existing files for editing.

  Default command key: **Ctrl+O**

  Tool bar action:

  Desktop mouse action: **Click Right Button**

Selecting this item shows you a standard <u>dialog</u> box, from which you can specify the file or files you want to work on.

   <u>About Opening Existing Files</u>

File View

Opens one or more existing files in *read only* mode. You will be able to examine their contents, but not alter them.

    Default command key: None

    Tool bar action: **Shift +**
    Desktop mouse action: **Shift + Click Right Button**

Selecting this item starts a <u>dialog</u> that is similar to the one produced by the **File Open** command. However, this dialog does not have a **Read Only** check box.

You can remove *read only* mode from a file with the <u>status bar</u> or with the **Options Current Modes** command.

        About Opening Existing Files

File Insert

Inserts an existing file anywhere into the text you're editing.

Default command key: None

Selecting this item shows you a standard <u>dialog</u> box, from which you can specify the file to insert.

<u>About Inserting An Existing File</u>

File Save

Saves the contents of a file you're editing to a disk file.

Default command key: **Ctrl+S**

Tool bar action:

If you haven't <u>associated a name</u> with the file yet, PFE treats this menu item as if you'd selected the **File Save As** command instead.

You cannot use this option to save a template that you're editing into a template file. Instead, use the **Template Store** or **Template Store As** commands.

If the **File Modes** of the current file so specify, and if a file of the same name already exists, PFE will maintain a <u>backup copy</u>, allowing you quickly to revert to a previous version of the file.

About Saving A File To Disk

File Save As

Saves the contents of a file to disk, allowing you to specify the file name.

Default command key: None

Tool bar action:

Selecting this item shows you a standard <u>dialog</u> box, from which you can specify the name of the disk file you want to save the data to.

Using this option changes the name associated with the window, so that future **File Save** commands will write to the same disk file automatically.

The dialog allows you to specify that, if you save to a file of the same name as one that already exists, a <u>backup copy</u> is kept, allowing you quickly to revert to a previous version of the file.

About Saving A File To Disk

File Save All

Saves all altered files, stores all altered templates into template files, and saves all altered template files in a single operation.

Default command key: None

PFE will check through all the files and templates that you currently have open, saving them if they have altered.

The process is done in several stages. First, all altered named files are written to disk; then all altered templates that are already associated with template files are stored in the in-memory copy of the relevant template files.

Next, PFE will ask you if you want to save each altered file that does not have a name. You can elect not to save a file if you wish; if you do want to save it, you'll see a dialog asking you to give a file name.

After this, PFE will ask you if you want to store each unnamed template. If you elect to store one, you'll be asked to supply a name for the template, and to specify the template file it is to be stored in. If you don't have any template files attached, you'll see a warning message, and this step will be skipped.

Finally, PFE will save all the attached template files that have been altered to disk.

Clicking **Cancel** in any of the dialogs that occur in this process will cancel the entire operation.

About Saving A File To Disk

File Write

Writes the contents of a file to disk,   allowing you to specify the file name

    Default command key: None

Selecting this item shows you a standard <u>dialog</u> box, from which you can specify the name of the disk file you want to write the data to.

The dialog allows you to specify that, if you write to a file of the same name as one that already exists, a <u>backup copy</u> is to be kept, allowing you quickly to revert to a previous version of the file.

Unlike the **<u>File Save As</u>** command, this command does *not* permanently <u>associate</u> the name of the disk file with the window showing the data.

      <u>About Saving A File To Disk</u>

File Name

Changes the disk file name associated with a window.

    Default command key: None

Selecting this item shows you a standard <u>dialog</u> box, from which you can specify the name of the disk file you want to associate with the window.

After using this command, the **<u>File Save</u>** command will write the data to disk in a file with the name you've specified. The file is marked as having been altered, so that the   **<u>File Save</u>** command will be immediately available for you to save the file with its new identity

This command does not affect any existing disk file.

<u>About Saving A File To Disk</u>

File Mail

Mails the data in the current window to someone else, using a <u>MAPI</u>-compliant mailer.

    Default command key: None

The command saves the data in the current window to a temporary file, then calls the mail system to prompt you for delivery instructions. The file will be mailed as an attachment.

Since the mailer may itself need to make a further copy of the file, you should ensure that you have adequate disk space available for this operation.

You will not be able to select this option if PFE does not detect a MAPI-compliant mailer in your system configuration when you start it.

<u>About Mailing A File</u>

File Close

Closes the file being shown in the current window. The current window, and all other windows showing the same file, will be closed.

Default command key: None

If you have changed the file concerned, and haven't saved the changes to disk, PFE will prompt you and allow you to save the changes, to discard them, or to cancel the close.

About Closing A File

File Close All

Closes all the files that you're working on. All windows will be deleted.

Default command key: None

If you have changed any of the files concerned, and haven't saved the changes to disk, PFE will prompt you and allow you to save the changes, to discard them, or to cancel the entire close operation.

About Closing A File

File Print

Prints all or part of the current file.

Default command key: **Ctrl+P**

Tool bar action:

The command starts a <u>dialog</u> that allows you to specify the details of how the file is to be printed and on which printer.

By default, PFE uses the same printer that you specified the last time you printed a file; default details of the printing, such as line folding and page headers, are taken from the <u>window modes</u> set on the current window.

You can also change the printer setup and the font to be used.

<u>About Printing A File</u>

File Print Setup

Selects the printer to be used by default for printing files, alters its setup details, and selects the font to be used.

Default command key: None

Tool bar action: **Shift +**

The command starts a <u>dialog</u> that allows you to configure the various details. The settings are recorded, and will be used the next time you start PFE.

About Printing A File

File Exit

Terminates your PFE session

    Default command key: None

This command will end your PFE session. If any of the files you're working with have changed, and you haven't saved the changes to disk, PFE will prompt you for each one in turn. You can choose to save the changes, discard them, or cancel the closedown.

The **Exit Windows** item in PFE's system menu allows you to end your PFE session *and* terminate Windows itself in a variety of ways.

About Ending a PFE Session

File Most-Recently-Used List

This is not a single menu item, but a list that PFE adds automatically to the end of the **File** menu.

Whenever you open a file, PFE records the name in the list, ordering it so that the files you have used most recently always appear at the top. You can then re-open any file in the list simply by clicking on the menu item.

The file will be opened for editing (unless the copy on disk is marked as read only). You can force PFE to open the file in read only mode by holding the **Control** key down as you select the menu item.

Normally, when you select a file from this list, PFE will create the window that shows it at the same size and position as when you last used the file. The caret will automatically be moved to the position in was at when you closed the file. You can suppress this behaviour, so that the window size and position will be set to default values, and the caret will be set to the start of the file, by holding the **Shift** key down as you select the menu item

Unless you started PFE in multi-instance mode with a command line option or with the **Options Preferences** command, PFE records the list in its initialisation file whenever you exit it. It will always re-instate the list when you start the next session.

You can configure how PFE re-uses saved window positions, and the size and display details of the list with the **Options Preferences** command.

When the list contains more files than can be displayed on the **File** menu, an extra menu command **More Files** is automatically added (if you've set the number of files you want shown on the menu to zero, the menu command will be **Recent Files**). The command starts a dialog that shows you the entire contents of the list, and lets you pick the file you want to open.

The Edit Menu

This item on the menu bar contains options concerned with altering text. It will not be shown if you have no files open.

Edit Undo
Edit Clear Undo
Edit Cut
Edit Copy
Edit Paste
Edit Select All
Edit Select Word
Edit Delete Line
Edit Delete To End Of Line
Edit Show Caret
Edit Goto Line
Edit Find
Edit Replace
Edit Repeat Last Find
Edit Repeat Last Replace
Edit Text

Edit Undo

Undoes the last edit action, returning the current file to its previous state

    Default command key: **Ctrl+Z**

    Tool bar action:

The text of this command on the **Edit** menu is not constant, but changes to indicate the nature of the action that will be undone if you use it. For example, if the last edit action you performed was to cut an area of text to the clipboard, the menu command will be **Edit Undo Cut**. Similarly, **Edit Undo Drag-Drop** indicates that the last edit action you performed was a drag-and-drop move or copy of text, and so on.

When there is no edit action to undo, or PFE is unable for some reason to reverse the last change you made, the menu command will show **Can't Undo**, and you will not be able to select the command.

Normally PFE records the last 32 edit actions that you have performed on each file that you are editing. You can change this to record more or fewer actions with the **Options Preferences** command.

        About Undoing Edit Actions

Edit Clear Undo

Clears the recorded details of all the edit actions performed up to this point for the current file, returning the memory used to the system

    Default command key: None

This command allows you to free the memory that PFE is using to record the changes you have made to the current file. Depending on what edit changes you have made to the current file, a significant amount of memory could be freed for other uses.

Once you have used this command, you will be unable to undo any preceding edit changes to the current file.

The command will ask you to confirm that you wish to take this action before carrying it out.

Normally PFE will automatically perform this action when you write a file to disk. You can change this behaviour, so that undo actions are discarded only when you explicitly command it with the **Options Preferences** command.

        About Undoing Edit Actions

Edit Cut

Deletes the highlighted text in the current window, placing it on the clipboard.

Default command key: **Ctrl+X**

Tool bar action:

After this operation, you will be able to paste the deleted text into another PFE window, or into another application

About Cutting, Copying and Pasting Text

Edit Copy

Copies the highlighted text in the current window onto the clipboard.

Default command key: **Ctrl+C**

Tool bar action:

After this operation, you will be able to paste the deleted text into another PFE window, or into another application.

Normally the text you've copied remains highlighted after this operation. You can arrange for the highlight to be removed with the **Options Preferences** command.

About Cutting, Copying and Pasting Text

Edit Paste

Pastes the contents of the clipboard into the current window at the position of the caret

    Default command key: **Ctrl+V**

    Tool bar action:

You will only be able to perform this operation if the clipboard contains data in text format. PFE does not support pasting of bitmaps or other clipboard formats.

About Cutting, Copying and Pasting Text

Edit Select All

Highlights the entire contents of the current window.

    Default command key: None

You can perform this operation wherever the caret currently is in the file. After it the caret will be positioned at the end of the file.

<u>About Selecting Text</u>

Edit Select Word

Highlights the <u>word</u> that the caret is currently in.

      Default command key: None
      Mouse shortcut: Double-click the left button

          <u>About Selecting Text</u>

Edit Delete Line

Deletes the entire line containing the caret.

Default command key: **Ctrl+Shift+K**

This command deletes the entire line, wherever the caret is within it. The caret moves to the start of the following line.

About Deleting Text

Edit Delete to End of Line

Deletes the text from the position of the caret to the end of the line containing it

Default command key: **Ctrl+K**

If the caret is at the start of a line that is completely empty, the line following is moved up to close the gap.

About Deleting Text

Edit Show Caret

Moves the data being shown in the current window so that the caret is visible, and the line containing it is as close as possible to the middle of the window

     Default command key: **Ctrl+F5**

Edit Goto Line

Moves the caret to a specific position within the file.

Default command key: **Ctrl+G**
Used from status bar?:   **Yes**

The command starts a <u>dialog</u> that allows you to specify the position within the file to move to. You can also choose to highlight all the text between the current position of the caret and the target line.

You can move to a specific line, or to a specific character location within a specific line.

<u>About Moving to Specific Lines</u>

Edit Find

Searches the current window for a text string.

Default command key: **F2**

Tool bar action:

The command starts a standard <u>dialog</u> that lets you specify the details of the search. The dialog remains visible, and can be re-used for another search in the same window, or in another window.

Any highlighted text in the current window will be set up as the default string to be searched for.

<u>About Finding Text</u>

Edit Repeat Last Find

Repeats the last search operation, without prompting for details.

Default command key: **Shift+F2**

Tool bar action: **Shift +**

The command repeats the last search operation, using the details last set with the **<u>Edit Find</u>** command.

If you have not previously performed a search operation, the command acts as the **Edit Find** command and shows the standard dialog.

If the **Edit Find** <u>dialog</u> is visible, the command activates it rather than performing a search.

<u>About Finding Text</u>

Edit Replace

Searches the current window for a text string, and optionally replaces occurrences with another string.

Default command key: **F3**

Tool bar action:

The command starts a standard <u>dialog</u> that lets you specify the details of the operation. The dialog remains visible, and can be re-used for another search in the same window, or in another window.

Any highlighted text in the current window will be set up as the default string to be searched for.

Using the dialog, you can either scan through the file, deciding whether or not to replace each occurrence of a match; or you can elect to replace all occurrences automatically.

<u>About Replacing Text</u>

Edit Repeat Last Replace

Repeats the last replace operation, without prompting for details.

Default command key: **Shift+F3**

Tool bar action: **Shift +**

The command repeats the last replace operation, using the details last set with the **<u>Edit Replace</u>** command.

If you have not previously performed a replace operation, the command acts as the **Edit Replace** command and shows the standard dialog.

If the **Edit Replace** <u>dialog</u> is visible, the command activates it rather than performing a replacement.

<u>About Replacing Text</u>

Edit Text

Shows a popup sub-menu containing text-related commands.

<u>Edit Text Transpose Characters</u>
<u>Edit Text Uppercase Selection</u>
<u>Edit Text Lowercase Selection</u>
<u>Edit Text Indent</u>
<u>Edit Text Undent</u>
<u>Edit Text Insert ASCII Code</u>
<u>Edit Text Match Brace</u>
<u>Edit Text Match Brace Select</u>
<u>Edit Text Widen Brace Select</u>

Edit Text Transpose Characters

Transposes the character to the right of the caret and the character to the left of the caret.

Default command key: None

It has no effect if the caret is positioned at the start or the end of a line.

<u>About Text Processing</u>

Edit Text Uppercase Selection

Changes all lower-case characters in the currently highlighted text to upper-case characters.

Default command key: None

About Text Processing

Edit Text Lowercase Selection

Changes all upper-case characters in the currently highlighted text to lower-case characters

Default command key: None

About Text Processing

Edit Text Indent

Moves the text in one or more lines right by one tab stop.

Default command key: None

If you do not have any text highlighted, PFE indents only the line that the caret is in. Otherwise, it indents all the lines containing highlighted text.

About Text Processing

Edit Text Undent

Moves the text in lines left by one tab stop.

    Default command key: None

If you do not have any text highlighted, PFE undents only the line that the caret is in. Otherwise, it undents all the lines containing highlighted text.

Lines that do not start with white space are not affected.

<u>About Text Processing</u>

Edit Text Insert ASCII Code

Inserts a character specified by its ASCII code

    Default command key: None

The command starts a <u>dialog</u> that lets you specify the ASCII code of the character to insert. You can choose from a preset list of control characters (ASCII codes 1 to 31) or type the actual decimal code value yourself.

       <u>About Text Processing</u>

Edit Text Match Brace

Moves the caret to a matching brace character

     Default command key: **Ctrl+B**

The action of this command depends on the language type defined in the current window's <u>window modes</u>.

Language type "(none)" or "TeX"

The command checks that the caret is positioned on a character that is either an opening brace from the set **{ [(** or **<**, or is a closing brace from the set **}])** or **>**.

If so, PFE scans the window in the appropriate direction to locate the brace that matches it, having regard to nested brace pairs. If a match is found, the caret is moved to lie on the caret; if no match is found, the caret remains where it is.

In scanning for the matching brace character no regard is paid to any syntactic elements of any kind in the text.

Language type "C"

The command checks that the caret is positioned on a character that is either an opening brace from the set **{ [** or **(**, or is a closing brace from the set **}]** or **)**.

If so, PFE scans the window in the appropriate direction to locate the brace that matches it, having regard to nested brace pairs. If a match is found, the caret is moved to lie on the caret; if no match is found, the caret remains where it is.

In scanning for the matching brace character regard is paid to some elements of C language syntax. Characters within comments, or inside pre-processor directives, are not scanned; nor are characters defined as character literals. However, depending on where the caret initially lies in relation to syntactic elements, on some occasions it may not be possible to make an unambiguous determination of the correct matching brace. This will happen especially if the syntactic elements in the text are invalid or not yet complete.

If it is required to both move to a matching brace *and* highlight the text between the braces, you can use either the **<u>Edit Text Match Brace Select</u>** or **<u>Edit Text Widen Brace Select</u>** commands.

     <u>About Text Processing</u>

Edit Text Match Brace Select

Moves the caret to a matching brace character, highlighting all the text between and including the brace characters

  Default command key: **Ctrl+Shift+B**

The action of this command depends on the language type defined in the current window's <u>window modes</u>.

Language type "(none)" or "TeX"

  The command checks that the caret is positioned on a character that is either an opening brace from the set **{ [(** or **<**, or is a closing brace from the set **}])** or **>**.

  If so, PFE scans the window in the appropriate direction to locate the brace that matches it, having regard to nested brace pairs. If a match is found, the caret is moved to lie on the caret and all the text between and including the brace characters is highlighted; if no match is found, the caret remains where it is.

  In scanning for the matching brace character no regard is paid to any syntactic elements of any kind in the text.

Language type "C"

  The command checks that the caret is positioned on a character that is either an opening brace from the set **{ [** or **(**, or is a closing brace from the set **}]** or **)**.

  If so, PFE scans the window in the appropriate direction to locate the brace that matches it, having regard to nested brace pairs. If a match is found, the caret is moved to lie on the caret and all the text between and including the braces is highlighted; if no match is found, the caret remains where it is.

  In scanning for the matching brace character regard is paid to some elements of C language syntax. Characters within comments, or inside pre-processor directives, are not scanned; nor are characters defined as character literals. However, depending on where the caret initially lies in relation to syntactic elements, on some occasions it may not be possible to make an unambiguous determination of the correct matching brace. This will happen especially if the syntactic elements in the text are invalid or not yet complete.

If it is required to both move to a matching brace *without* highlighting the text between the braces, use the **Edit Text Match Brace Select** command.

   About Text Processing

Edit Text Widen Brace Select

Highlights the next largest area of text delimited by matching brace characters and including either the character under the caret, or the current selection.

Default command key: **Ctrl+Shift+W**

The action of this command depends on the language type defined in the current window's window modes.

Language type "(none)" or "TeX"

The command attempts to locate a matching pair of brace characters that contain either the character under the caret, or the whole of any highlighted text. It regards opening braces as any character from the set **{ [(** or **<**, and closing braces as the corresponding characters from the set **}])** or **>**.

If a matching brace pair is found, the text between and including them is highlighted.

In scanning for the matching brace characters no regard is paid to any syntactic elements of any kind in the text.

Language type "C"

The command attempts to locate a matching pair of brace characters that contain either the character under the caret, or the whole of any highlighted text. It regards opening braces as any character from the set **{ [** or **(**, and closing braces as the corresponding characters from the set **}]** or **)**.

If a matching brace pair is found, the text between and including them is highlighted.

In scanning for the matching brace characters regard is paid to some elements of C language syntax. Characters within comments, or inside pre-processor directives, are not scanned; nor are characters defined as character literals. However, depending on where the caret initially lies in relation to syntactic elements, on some occasions it may not be possible to make an unambiguous determination of the correct matching brace. This will happen especially if the syntactic elements in the text are invalid or not yet complete.

Repeated use of this command will have the effect of selecting increasingly larger sections of text, enabling you easily to locate and access large delimited blocks.

About Text Processing

The Options Menu

This item on the menu bar contains various system configuration options. Not all the items will appear if you don't have a file open.

Options Current Modes
Options Default Modes
Options Reset Modes
Options Key Mapping
Options Preferences
Options DDE Server

Options Current Modes

Starts a <u>dialog</u> that allows you to configure the modes that apply to the current window, and to the file that is being shown in the current window.

Default command key: None

<u>About File and Windows Modes</u>

Options Default Modes

Starts a <u>dialog</u> that allows you to configure the modes that PFE applies to windows and files when you open existing files or create new ones.

    Default command key: None

          <u>About File and Windows Modes</u>

Options Reset Modes

Resets the window modes that apply to the current window, and the file modes that apply to the file it's showing, to the default values.

    Default command key: None

The mode values applied are derived from values defined by the **Options Default Modes** command.

       About File and Windows Modes

Options Tool Bar Hide

Removes the tool bar from the screen.

Default command key: None

About The Tool Bar

Options Tool Bar Show

Makes the tool bar visible on the screen in its last position.

Default command key: None

About The Tool Bar

Options Tool Bar Top

Sets the tool bar to appear as a bar along the top edge of the main PFE window.

Default command key: None
Tool bar action: **Click + Drag**

About The Tool Bar

Options Tool Bar Bottom

Sets the tool bar to appear as a bar along the bottom edge of the main PFE window.

Default command key: None
Tool bar action: **Click + Drag**

About The Tool Bar

Options Tool Bar Left

Sets the tool bar to appear as a bar down the left edge of the main PFE window.

Default command key: None
Tool bar action: **Click + Drag**

About The Tool Bar

Options Tool Bar Right

Sets the tool bar to appear as a bar down the right edge of the main PFE window.

Default command key: None
Tool bar action: **Click + Drag**

About The Tool Bar

Options Tool Bar Floating

Sets the tool bar to appear as a floating window that you can position anywhere within the main PFE window.

     Default command key: None
     Tool bar action: **Click + Drag**

A floating tool bar window always remains visible above any edit windows.

<u>About The Tool Bar</u>

Options Key Mapping

Maps keys to functions.

    Default command key: None

The command starts a <u>dialog</u> that enables you to set which operations are to be performed when you press particular keys. You can save the mappings you specify so that PFE will use them automatically the next time you start it. You can also save them to as individual key mapping files, which you can load for specific purposes as you require.

By default, PFE uses a set of <u>built-in key mappings</u>.

<u>About Key Mapping</u>

Options Preferences

Configures PFE to suit your requirements

Default command key: None

This command allows you to configure PFE so that it works in the way that suits you best. It runs a dialog that lets you set the various configuration options.

Options DDE Server

Turns the DDE server facilities on and off.

    Default command key: None

This command can be used at any time to start and stop PFE's DDE server system. Normally, PFE will act as a DDE server by default, but if you've configured the system to run multiple instances by default with the **Options Preferences** command, or started PFE with a "**/m**" command line option, the DDE server will not be running.

The menu item will be checked if the DDE server is running. You should normally try to have only one instance of PFE that is a DDE server running at any time, as it's not predictable which one another application will connect to.

Controlling PFE Over A DDE Link

The Template Menu

This menu item contains options that relate to handling templates. Not all the options will be shown if you don't have any files open.

Template Attach File
Template Detach File
Template Create File
Template Save File
Template Insert
Template Edit
Template New
Template Store
Template Store As
Template Delete
Template Insert Mark
Template Find Mark

About Templates

Template Attach File

Attaches a template file, loading into memory so the templates within it can be used.

Default command key: None

The command starts a dialog that allows you to browse the files on your disk for template files, which have type "**.tpl**".

You can arrange for PFE automatically to attach the template files you use frequently, and also automatically to attach one template file specific to the directory you start it in.

About Templates

Template Detach File

Detaches a template file, unloading it from memory.

    Default command key: None

The command starts a <u>dialog</u> that allows you to specify which attached template files you wish to detach. Once a template file is detached, you cannot use the templates it contains.

You can't detach a template file if you're currently editing any of the templates it contains. You will need to save any changes you want to keep with the **<u>Template Store</u>** or **<u>Template Store As</u>** commands, and then close the windows.

      <u>About Templates</u>

Template Create File

Creates an empty template file and automatically attaches it.

    Default command key: None

The command starts a <u>dialog</u> that allows you to choose the name of the template file to create. This should have a file type of "**.tpl**". PFE will create the file on disk and initialise it.

Template files do not contain pure text, and this is the only method by which you can create them.

<u>About Templates</u>

Template Save File

Saves a template file to disk after you have edited its contents

    Default command key: None

The command starts a <u>dialog</u> that shows you the names of the attached template files that have changed. You can choose which of them to save.

Template operations such as edit, delete, new and store work only the copy of the template file in memory. Any changes you make are not recorded permanently until you use this command.

      <u>**About Templates**</u>

Template Insert

Inserts a template from an attached template file into a window.

Default command key:	**F9**
Tool bar action:	

The command starts a <u>dialog</u> that shows you the names of the attached template files, and the templates they contain. You can select any template from any file, then click **OK** to action the insert.

The template text is inserted into the current window at the position of the caret. Unlike the case of inserting a file from disk, the caret is left positioned at the *start* of the inserted text: this allows you to use the **<u>Template Find Mark</u>** command to fill in the positions marked as needing text input.

<u>About Templates</u>

Template Edit

Edits a template contained in an attached template file.

Default command key: None

Tool bar action: **Ctrl +**

The command starts a <u>dialog</u> that shows you the names of the attached template files, and the templates they contain. You can select any template from any file, then press **OK** to edit it.

PFE will create a new window, and copy the text of the template into it. You can edit the template as you wish; then, when you've made the changes, you can use the **<u>Template Store</u>** command to update the in-memory copy of the template file.

Note that any changes you make will not be recorded permanently in the disk copy of the template file until you use the **<u>Template Save File</u>** command.

<u>About Templates</u>

Template New

Creates a new, empty window in which you can type the text of a new template.

Default command key: None

Tool bar action: **Shift +**

The window created is not associated with either a template name, or a template file. To record what you've typed, you will need to use the **Template Store As** command to name the template and place it in a specific template file.

About Templates

Template Store

Stores a template that you've been editing into the in-memory copy of the template file that it is associated with.

Default command key: None

The previous version of the template that the template file contains will be overwritten by this command. You can store a maximum of 64 templates in a template file, and the total space used by all templates in the file cannot exceed a total of approximately 62 kilobytes.

Note that the template file on disk is not altered by this command. To save your changes permanently, you next need to use the **Template Save File** command.

If you created the window you're editing the template in with the **Template New** command, and haven't yet stored the template in a template file, this command will act like the **Template Store As** command.

About Templates

Template Store As

Stores a template that you've been editing into the in-memory copy of a template file, allowing you to specify both the template file, and the name of the template.

Default command key: None

The command starts a dialog that lets you specify the details of the operation. You should first select which of the attached template files you want to store the template file in, then specify the template name. The total space used by all templates in the file cannot exceed a total of approximately 62 kilobytes.

You can select the name of an existing template if you wish; as this will overwrite the previous contents, PFE will ask you to confirm that you really want to do this.

Note that the template file on disk is not altered by this command. To save your changes permanently, you next need to use the **Template Save File** command.

About Templates

Template Delete

Deletes templates from the in-memory copy of any attached template file

    Default command key: None

The command starts a <u>dialog</u> that lets you specify the details of the operation. You should first select which of the attached template files you want to delete templates from, then select one or more template names from the list of those it contains.

Note that the template file on disk is not altered by this command. To save your changes permanently, you next need to use the **<u>Template Save File</u>** command.

       <u>About Templates</u>

Template Insert Mark

Inserts a *template mark* into a template that you're editing

    Default command key: **F6**

A template mark is a sequence of characters ("**<???>**") that you can use to mark positions in a template where you need to put variable details after you've inserted it.

When you insert a template into a file, you can use the **<u>Template Find Mark</u>** command to quickly move to all the template marks in turn and replace them with whatever is appropriate.

       <u>About Templates</u>

Template Find Mark

Moves the caret to the next *template mark* in a file or a template and highlights it

    Default command key: **F4**

A template mark is a sequence of characters ("**<???>**") that you can use to mark positions in a template where you need to put variable details after you've inserted it.

When you insert a template into a file, you can use this command to quickly move to all the template marks in turn and replace them with whatever is appropriate.

<u>About Templates</u>

The Execute Menu

This menu contains items that relate to running DOS commands and starting Windows applications.

Execute DOS Command to Window
Execute Repeat DOS Command to Window
Execute DOS Prompt
Execute Launch Application
Execute Launch Windows Tool
Execute Configure Windows Tools
Execute Control Panel
Execute File Manager
Execute Print Manager
Execute Program Manager
Execute Task Manager

About Running DOS Commands
About Launching Windows Applications
About Windows Tools
About Keyboard Macros

Execute DOS Command To Window

Executes a DOS application, such as a compiler or a built in command, and captures the output in a window.

>   Default command key: **F11**

>   Tool bar action:

The command starts a <u>dialog</u> that lets you give the command line to be executed. It can be any DOS command, including built-in ones like "**dir**". You can also automatically <u>substitute</u> parts of the name of the file showing in the current window into the command line.

You can specify the working directory that the command is to run in. This affects only the DOS command - PFE itself will still use its previous working directory.

When you start the dialog, PFE sets the command string and the proposed working directory name to the values you set the last time you used it in this or a previous session.

You may execute only one DOS command at a time. Until the command completes, PFE will reject attempts to start another.

This command is not available if you are running the 16-bit Edition under Windows NT, or the 32-bit Edition under Windows 3.1 with Win32s.

>   **<u>About Running DOS Commands</u>**

Execute Repeat DOS Command To Window

Repeats the last DOS command you ran with the **Execute DOS Command To Window** command, capturing the command's output.

    Default command key: None

    Tool bar action: **Shift +**

PFE will repeat the last command exactly, and will show you the command output in a window when it completes. If you have not previously run a command, PFE will act as if you selected the **Execute DOS Command To Window** command and will show you the dialog.

If the command line you specified in the dialog contained substitution points requesting that PFE automatically include parts of the file name of the file showing in the current window, these will be re-evaluated before the command is executed.

This command is not available if you are running the 16-bit Edition under Windows NT, or the 32-bit Edition under Windows 3.1 with Win32s.

       About Running DOS Commands

Execute DOS Prompt

Starts a separate task running the DOS command processor

Default command key: **Ctrl+F11**

Tool bar action:

PFE starts a new command processor task, which runs completely independently from it.

If you have defined a command processor with the the **Options Preferences** command, this program will be run.

If not, PFE will look at the environment variable **COMSPEC**, and if this is defined, will run the program named in it.

If neither of the above options specifies a command processor, PFE will start **command.com** (16-bit Edition) or **cmd.exe** (32-bit Edition)

About Running DOS Commands

Execute Launch Application

Launches a DOS or Windows application as a task running independently of PFE

    Default command key: **Shift+F11**

    Tool bar action:

The command starts a <u>dialog</u> that lets you give the command line to be executed. It can specify any DOS or Windows application.

You can specify the working directory that the application is to run in. This affects only the application - PFE itself will still use its previous working directory.

When you start the dialog, PFE sets the command string and the proposed working directory name to the values you set the last time you used it in this or a previous session.

Applications launched by this command run totally independently of PFE, and their output is *not* captured. To run something like a compiler, whose output you want to catch in an edit window, use the **<u>Execute DOS Command To Window</u>** command.

       <u>About Launching Windows Applications</u>

Execute Launch Windows Tool

Launches one of the Windows applications configured in the list of Windows Tools

    Default command key: **F12**

    Tool bar action: **Shift +**

This command gives a quick way of starting a pre-defined Windows application such as the Dialog Editor or a debugger.

When you select it, PFE shows you a <u>dialog</u> that gives a list of the configured Windows tools. You can select one from the list, modify the associated command line and working directory if you wish, and press **OK** to launch it.

Applications launched in this way run independently of PFE, and their output is *not* captured. To run something like a compiler, whose output you want to catch in an edit window, use the **<u>Execute DOS Command To Window</u>** command.

To set up or edit the list of available Windows tools, use the **<u>Execute Configure Windows Tools</u>** command.

       <u>About Windows Tools</u>

Execute Configure Windows Tools

Configures details of the defined Windows Tools.

Default command key: **Shift+F12**

Tool bar action: **Ctrl +**

This command starts a <u>dialog</u> that lets you add or delete entries from the list of Windows tools, or modify existing ones.

<u>About Windows Tools</u>

Execute Control Panel

Starts the Windows Control Panel application.

Normally PFE will execute the standard Control Panel from **control.exe**. However, you can specify the command line to be executed with the **<u>Options Preferences</u>** command.

Execute File Manager

Starts the Windows File Manager application.

Normally PFE will execute the standard File Manager from **winfile.exe**. However, you can specify the command line to be executed with the **Options Preferences** command.

Execute Print Manager

Starts the Windows Print Manager application.

Normally PFE will execute the standard Print Manager from **printman.exe**. However, you can specify the command line to be executed with the **Options Preferences** command.

Execute Program Manager

Starts the Windows Program Manager application.

Normally PFE will execute the standard Program Manager from **progman.exe**. However, you can specify the command line to be executed with the **Options Preferences** command.

Execute Task Manager

Starts the Windows Task Manager application.

Normally PFE will execute the standard Task Manager from **taskman.exe**. However, you can specify the command line to be executed with the **Options Preferences** command.

The Macro Menu

This menu item contains commands relevant to handling keyboard macros, which allow you to record sequences of keystrokes and menu commands and replay them.

<u>Macro Start Recorder</u>
<u>Macro Stop Recorder</u>
<u>Macro Replay</u>

Macro Start Recorder

Starts recording your key presses and menu selections as a keyboard macro, which you can then replay

    Default command key: **Shift+F7**

    Tool bar action:

Once you've selected this command, PFE records all the keys you press, and all the menu selections you make, in its keyboard macro buffer. When you've performed all the actions you want recorded, use the **Macro Stop Recorder** command to turn recording off.

You can record a maximum of 1024 key strokes or menu actions. Two-character command sequences count as a single stroke.

Things that you type in dialog boxes are not recorded, nor are mouse actions.

Once a keyboard macro has been recorded, you can replay it as often as you wish with the **Macro Replay** command.

In this version of PFE, any previously stored keyboard macro is deleted when you turn recording on. Also, you cannot save the actions you record permanently.

       About Keyboard Macros

Macro Stop Recorder

Stops recording your key presses and menu selections as a keyboard macro

Default command key: **Ctrl+F7**

Tool bar action:

Once you've selected this command, PFE no longer records the keys you press, and the menu selections you make, in its keyboard macro buffer.

Once a keyboard macro has been recorded, you can replay it as often as you wish with the **Macro Replay** command.

About Keyboard Macros

Macro Replay

Replays the current keyboard macro

    Default command key: **F7**

    Tool bar action: **Shift +**

The command causes PFE to replay all the key presses and menu selections that you've recorded. If any operation causes an error - for example, you might replay a **File Save** command when no file is open - the replay is terminated.

PFE does not record what you type in dialog boxes; thus, if any recorded key starts a dialog, you will need to fill in the details whenever you replay the macro. Pressing the **Cancel** button in any dialog will cause PFE to end the macro replay.

       About Keyboard Macros

The Window Menu

This menu item contains commands relevant to handling edit windows. It does not appear if you have no files open.

Window Tile Horizontal
Window Tile Vertical
Window Cascade
Window Iconize All
Window Arrange Icons
Window Select
Window Next
Window Duplicate
Window Close
Window Widen

The Window List
About Window Manipulation

Window Tile Horizontal

Arranges all the non-iconic edit windows in a tile pattern, placing them side by side so that each is as deep as possible.

Default command key: **Shift+F4**

<u>About Window Manipulation</u>

Window Tile Vertical

Arranges all the non-iconic edit windows in a tile pattern, placing them one above the other so that each is a wide as possible.

Default command key: None

About Window Manipulation

Window Cascade

Arranges all non-iconic edit windows in a cascade pattern

Default command key: **Shift+F5**

About Window Manipulation

Window Iconize All

Makes all edit windows into icons

Default command key: None

About Window Manipulation

Window Arrange Icons

Arranges all iconized edit windows so that their icons are visible, in rows at the bottom of the PFE desktop.

Default command key: None

Windows that are not icons are not affected by this command

<u>About Window Manipulation</u>

Window Select

Allows you to choose between many open windows with a selection dialog

Default command key: **Ctrl+W**

This command starts a <u>dialog</u> that lists the window captions of all the edit windows. You can refine the list to show only windows showing files that have associated names, windows showing un-named files, windows showing templates, or windows showing command output. You can also choose to see only windows that have been altered.

You can select the window you want to see from the dialog, and press the **OK** button to switch to it.

Window Next

Moves the focus to the next edit window that is not an icon.

Default command key: None

This command will not activate any window that is currently iconic. Windows are activated in the order in which PFE maintains their control information, which will not be the same as the ordering used by Windows itself.

About Window Manipulation

Window Duplicate

Makes an exact copy of the current window

    Default command key: None

The duplicate window created by this command shows the same file, with the caret in exactly the same place. All aspects of the new window are identical to the original.

You can produce as many duplicates of a window as you wish, and edit the same file in each of them,

      <u>About Window Manipulation</u>

Window Close

Closes the current window

    Default command key: None

If the window that you close is the only one showing that particular file, this command acts as if you'd used the **File Close** command. PFE will check whether the file has been changed, and will give you the opportunity of saving them.

       About Window Manipulation

Window Widen

Alters the size and horizontal position of the current window to make it as wide as possible within the PFE desktop

 Default command key: None

The window's vertical size and position are not affected

 About Window Manipulation

Window List

This is not a single menu item, but a list that is automatically added to the end of the **Window** menu whenever you open a window.

The list shows the window captions for up to 9 of the windows that you have open at the moment. You can switch to one of them simply by clicking the appropriate menu item.

If you have more than 9 windows open, a further command **More Windows** is added to the end of the list. This command is identical in effect to the **Window Select** command; it starts a dialog that lets you choose the window to activate.

The Help Menu

This menu lets you obtain help on using PFE, and also lets you run the Windows help engine on up to five help files of your own choice.

If you wish, you can always use the **Help Contents** command to enter the help system, and navigate through the information to find what you want. To help you locate things quickly, though, various other commands allow you to enter the help system at specific points; for example, the **Help Commands** command is useful if you want to look up a menu command to see what it does.

You can also obtain context-specific help when you're working in dialog boxes by either clicking on their **Help** buttons, or pressing the **F1** key. Many message boxes that ask you questions will also show you help if you press the **F1** key.

You can obtain interactive help on menu items and screen areas such as the tool bar and status bar by pressing **Shift+F1** or using the **Help Screen/Menu Help** command. This will let you point the mouse cursor at the menu item or screen area you want help on, and click the left mouse button to enter the help system.

The **Help Context Help** command allows you to enter a specific Windows help file using the text highlighted in the current window as a search key. This is a convenient way of obtaining help on, for example, Windows API routines.

> **Help Contents**
> **Help Search For Help On**
> **Help Screen/Menu Help**
> **Help How To Use Help**
> **Help Commands**
> **Help Procedures**
> **Help Reference**
> **Help Context Help**
> **Help About**
>
> **User-Configurable Items**

Help Contents

Starts the Windows help engine to give help on PFE. The help file is opened at the Contents page

Default command key: **F1**

Help Search For Help On

Starts a dialog that allows you to search the PFE help file for a given keyword

    Default command key: None

You can type the keyword you want to search for, or you can select one of the ones shown in the list, then click the **Show Topics** button to see all the help topics that refer to that keyword.

You can then select the topic that interests you, and open the help file to show it by clicking the **Goto Topic** button

Help Screen/Menu Help

Enters the interactive help system that lets you point the mouse cursor at the menu item or screen area you want help on

    Default command key: **Shift+F1**

When you use this command, the mouse cursor will change to become an arrow-and-question mark, to show that interactive help is active.

For help on an area of the status bar or on a tool bar button, point the mouse cursor at the appropriate part of the screen and click the left button once. For help on a menu item, select the item as you would do actually to use the command, and click the left button.

PFE will start the Windows help engine, and will show the topic appropriate to where you clicked the mouse button.

If you decide you don't want interactive help after using the command, you can return the mouse cursor to normal by pressing the **Escape** key.

When interactive help is operating, any keys you press will be ignored, and the tool bar, status bar and menu items will not have their normal effect.

Help How To Use Help

Starts the Windows help engine to give you help on how the help system works

    Default command key: None

Help Commands

Starts the Windows help engine to give help on PFE's menu commands.

    Default command key: None

Help Procedures

Starts the Windows help engine to give help on how you perform various tasks using PFE.

    Default command key: None

Help Reference

Starts the Windows help engine to give help on PFE. The help file is opened at the Reference section, which gives you such items as the list of functions that you can bind to keys, and so on

   Default command key: None

Help Context Help

Starts the Windows help engine to give help associated with text in the current window

    Default command key: **Ctrl+F1**
    Mouse shortcut: Double-click the right button

If you have some highlighted text in the current window, PFE will pass the first 64 characters of it (with leading and trailing spaces removed, and non-printable characters converted to spaces) to the Windows help engine as a *partial key*.

With no text highlighted, PFE first performs an **Edit Select Word** command to select the entire word containing the caret, then passes this to the help engine.

You can specify the name of the help file to be searched for context help with the **Options Preferences** command.

 If the help file contains an exact match for the text, you will see the appropriate help entry; if not, you will see a list of possible topics that might match.

Help About

Starts a dialog that gives you information about the particular version of PFE that you're using.

Default command key: None

Help User-Configurable Items

This is not a single **Help** menu item, but a list that you can define for yourself, so that you can access the help files that you want to work with quickly from PFE's own help menu.

PFE allows you to define up to five help files, and associate each with a menu item that is appended to the end of the **Help** menu, with the **Options Preferences** command.

Reference Not Available

Sorry, the material for this section is not yet complete

Procedures

This section of the help file contains overviews and descriptive material that will explain the features of PFE, and tell you the exact steps you need to take to achieve results.

In many cases, the descriptions will discuss running a dialog to perform an action. These topics will have links that take to step-by-step instructions for working with the dialogs; you can also access these instructions by clicking the **Help** button in the dialog itself, or by pressing the **F1** key.

Starting Up

Starting PFE

Running Single Or Multiple Instances
Specifying A Key Mapping File On The Command Line

Command Line Options

File Operations

Opening Existing Files From File Manager
Opening Existing Files From PFE
Creating A New File
Inserting An Existing File
Changing An Associated File Name
Saving A File To Disk
Closing A File
Mailing A File
Printing A File

Backing Up Existing Files
Working With Unix Format Files

Handling Text

Typing Text
Deleting Text
Selecting Text
Cutting, Copying and Pasting Text
Dragging And Dropping Text
Finding Text
Replacing Text
Text Processing Operations
Undoing Edit Actions
Templates
File And Window Modes

Moving About

Moving To Specific Lines
Finding Text

System Control

Window Manipulation
Screen Fonts

Starting PFE

Like any other Windows application, PFE can be started in a variety of ways.

- From **Program Manager** you can start it from an icon in a program group

- From **File Manager** you can start it by double clicking on its name. You can also start it up to edit one or more files by dropping their names onto the PFE program's name

You can also start it with the **File Run** commands of either of these applications; and from any other application that can launch programs.

However you start PFE, you can if you wish include on its command line a list of files that it is to open. If you do give a list, PFE will open each file in turn. Filenames given on the command line may include the normal DOS wildcard characters, and PFE will open all files whose names match. Files named on the command line will be opened for editing unless you also include the command line option "**/v**", when they will be opened in *read-only* mode.

By default, only one instance of PFE will run at a time. When you start a second one, it will pass control to the first, commanding it to open any files you've given on the command line.

If you wish, you can use the **Options Preferences** command to have PFE perform some file actions automatically as it starts up.   You can instruct to PFE create an empty, unnamed window, or show you the dialog for opening files when you start it with no file names on the command line.

Running Single Or Multiple Instances
Specifying A Key Mapping File On The Command Line

Command Line Options

Running Single Or Multiple Instances

By default, PFE attempts to run only one instance of itself at any time. When you start it for a second time, the first instance is detected, and it is activated. Any file names that you gave on the command line are passed to this first instance across a DDE link.

In many cases, you'll find this a convenient way to work, as it reduces the clutter on your screen, and you don't have to worry about which version of PFE is editing what file.

However, if you're working on several things at the same time, it can be distracting to have unrelated files being edited in the same instance of the editor, so PFE gives you the choice of running multiple copies of itself when you want.

INITIALISATION FILE CONFIGURATION

The easiest way to set PFE up to run with multiple copies of itself is to use the **Options Preferences** command to set this.

COMMAND LINE CONFIGURATION

You can also control how PFE starts by using command line options, which overrides the values set with the **Options Preferences** command. If you specify the "**/m**" option, PFE will always start a new instance; if you specify "**/s**", it will always try to activate a previous instance.

You may find it convenient to set up several icons in your Program Manager groups with different command line options. For example, you could have a general icon with a command line of simply

        pfe

to start PFE up for general editing. To edit your system files, you might have icons with command lines like:

        pfe /m c:\windows\win.ini

        pfe /m c:\autoexec.bat

and so on. These will start independent instances of PFE to edit these files, without getting in the way of any other editing work you're engaged in.

DDE CONSIDERATIONS

When you start PFE with the "**/m**" command line option, or if you don't use a command line option and have run-mode set to 1 in your initialisation file, the instance of PFE that starts will *not* run as a DDE server. This is so that any instance you start later with a "**/s**" switch will not activate this instance instead.

If you *want* an instance of PFE to run as a DDE server, you can either start PFE with a "**/s**" option (if run-mode is not set to 0) or use the **Options DDE Server** command to start any instance running a server.

Note that if you have more than one instance of PFE running as a DDE server, it's not predictable which of them will be connected to by another application. If you do use DDE, it's normally best to ensure that only one instance of PFE runs, or only one of several instances is acting as a server.

Specifying A Key Mapping File On The Command Line

By default, PFE will look for a file called **pfe.key** (if you're running the 16-bit Edition) or **pfe32.key** (if you're running the 32-bit Edition) in your Windows directory when it starts, and will load the <u>key mappings</u> that it contains. You can change the name of this default key mapping file with the **<u>Options Preferences</u>** command.

If you sometimes need to load key mappings from some *other* file on start-up, or to force PFE to use the built-in defaults described in this help file, you can use the "**/k**" <u>command line option</u> to name any key map file. For example, starting PFE with a command line

```
pfe /k c:\keymaps\uemap.key
```

would load the key mappings in the file **c:\keymaps\uemap.key**. If you have a key map file **pfe.key**, but want to use the built-in key mappings described in this help file instead, use the "/k" option and give the filename as the single character "**-**" (minus).

Note that when you use the "**/k**" option, the dialog started by the **<u>Options Key Mapping</u>** command will not save changed mappings to the file you specify, or to the standard key mapping file, by default. When saving you will need always to give an explicit file name.

Opening Existing Files From File Manager

If you have already started PFE running, you can pass files to it for editing from File Manager using the drag-and-drop technique. You can also do this from any other application that can act as a drag-and-drop server.

To open one or more files in this way, select them in the File Manager window; then click the left mouse button down while over one of the names and hold it down. Move the mouse cursor to anywhere in PFE's window and release the mouse button.

PFE will open each of the files in turn. They will all be opened such that you can alter them, except for files whose directory entries have the **read-only** attribute set, which will be opened in *read-only* mode.

Opening Existing Files From PFE

To start editing one or more existing files, use the **<u>File Open</u>** command. This starts a standard <u>dialog</u> that allows you to move around your directories to find the files you want to edit.

If you want to prevent yourself accidentally changing the files, click the **Read Only** box to tell PFE to open them all in *read-only* mode. Alternatively, you can use the **<u>File View</u>** command, which automatically does this. Whatever options you use, any file whose directory attributes include **Read Only** will always be opened in *read-only* mode.

Normally, the dialogs that open and save files will show you all the files that exist in each directory. You can arrange to set up your own selections by <u>configuring</u> the *filter list* that appears in the dialogs.

When each file is loaded, PFE looks at the *file type* to determine such things as the tab size to use, whether to treat the file as C language source, whether automatically to strip any final **Ctrl+Z** character, and so on. You can configure the settings it uses for different file types with the **<u>Options Default   Modes</u>** command.

If you select a file that is already open, PFE will not load a new copy from disk. Instead, it will activate one of the windows that are currently showing the file.

PFE will examine the contents of the file specified to see if it is a <u>template file</u>. If it is, it will be attached rather than opened for editing.

PFE can open files that are stored in the normal MS-DOS format (using CR-LF bytes to mark the end of each line) or in <u>Unix format</u> (using a single LF byte) automatically.

Configuring The File Type Filters

Whenever you use one of the **File Open**, **File View**, **File Insert**, **File Save As** or **File Write** commands, PFE will show you a standard Windows dialog that allows you choose the filename to use, moving around all your directories.

The possible file names in the current directory are shown in the list on the left; the files shown in this list are determined by your selection from the **List Files of Type** list at bottom left.

By default, there will be only one item in the list, which selects all files, using a pattern of "***.***". If you wished, you could configure this list to let you select a smaller group of files: for example, if you program in C, you might wish to see only files with types of "**.c**" and "**.h**" in one group, and files of type "**.dlg**" in another.

To set up the file filters that you require, use the **Options Preferences** command.

Creating A New File

The **File New** command opens a new, empty window in which you can type.

The window will not initially be associated with a file name. In order to save what it contains to disk, you'll need to use a command such as **File Save As**, which allows you to set the file name.

When the window is created, PFE applies the set of modes - that is, the tab size, wrap column, and other details - that are defined for "new files". You can use the **Options Current   Modes** command to change them for the new window; or you can change the default settings using the **Options Default Modes** command.

Inserting An Existing File

To insert an existing file into the window that you're editing, place the caret to the point at which you want the insertion to be made, then use the **File Insert** command.

This starts a standard dialog that allows you to move around your directories to find the file you want to insert. PFE can insert files that are in normal MS-DOS format (using CR-LF bytes to mark the ends of lines) or in Unix format (using single LF bytes) automatically.

PFE will insert the file into the window, leaving the caret positioned after the last character inserted.

By default, the dialog will show you all the files existing in each directory, but you can configure it to show you a restricted set.

If the file to be inserted is a template file, PFE will reject the operation.

Changing A File Name

When you edit a file, PFE will associate the name of the file with all the windows that show it. When you then use the **File Save** command to write the file to disk, PFE will use the associated name when it opens the file.

Some windows - such as those that you've created with the **File New** command, or those holding DOS command output produced by the **Execute DOS Command To Window** command - will not have file names associated with them. Or you may want to change the associated name to something else.

You can change the file name associated with a window, and save it to disk at the same time, by using the **File Save As** command. This runs a dialog that allows you to specify a file name; after saving the file, that file name is associated with all the windows showing the same file.

Alternatively, you can change the file name without writing to disk by using the **File Name** command. This also runs a dialog that lets you specify the associated file name, but makes no changes to your disk.

Whatever method you use to change the name of a file, PFE will examine the name and change the file and window modes that apply to the file, and to the current window, to match the new file extension.

Saving A File To Disk

To save a file to disk you have a choice of two commands.

- The **File Save** command will save the contents of the current window to disk, using the file name that's associated with the window. Any previous contents of the file will be lost.

- The **File Save As** command prompts you to supply the name of the disk file to write to. If you choose an existing file, PFE will warn you and ask if you really intend to overwrite it. The file name you give then becomes associated with the window.

If the window doesn't have a file name associated with it yet, both commands will ask for a file name. You can also associate a file name with a window at any time, without writing it to disk, using the **File Name** command.

If the current window contains a template, you won't be able to use the **File Save** or **File Save As** command. Instead, use the **Template Store** or **Template Store As** commands.

By default, you can't use the **File Save** command if you haven't changed the file. You can configure PFE to allow you to do this with the **Options Preferences** command.

When saving a file, PFE will normally preserve a backup copy of any existing file of the same name, so that you can quickly revert to a previous version of your work.

> **Setting The Format of the Saved File**
> **Backing Up Existing Files**

Setting The Format of a Saved File

PFE can save files using two ways of marking the end of each line:

- **DOS Format** writes the file so that each line is terminated by a carriage-return byte and a line-feed byte. This is the normal format for DOS files, and you should use this if you want the file to be readable by other DOS applications.

- **UNIX Format** writes the file so that each line is terminated by only a line-feed byte. This is the format used in UNIX systems; however, the file probably won't be usable with other DOS or Windows applications.

The format that PFE will use to save a file is shown in the status bar.

There are several ways to set the format that a file will be saved in:

- By default, PFE will detect that a file is in Unix format when it is opened, and will set the save format to "Unix".

- You can change between DOS and Unix formats by double-clicking the left mouse button in the format area of the status bar.

- You can use the **Options Current Modes** command to change the format of the current file.

- You can use the **Options Default Modes** command to set the format for files of specific types.

You can also control two aspects of what happens to the final line in the file when it is saved:

- You can choose whether or not the last data character in the file is followed by a line terminator. By default, PFE will always write a line terminator.

- You can choose to have PFE automatically add a **Ctrl+Z** character after the file data has been written. (If the last character in the file is *already* a **Ctrl+Z** character, PFE will not add a second)

You can determine the settings used for the current file with the **Options Current Modes** command, and can set the default values used for files of given file types with the **Options Default   Modes** command.

Working With Unix Format Files

Backing Up Existing Files

When you save a file to disk, it's often desirable to be able to keep a copy of any file of the same name that already exists. You can then quickly revert to a previous version of your work.

By default, all operations that write files to disk will perform a backup action. You can turn backup actions off completely, or configure the details of their operation, with the **Options Preferences** command.

One way of backing up files is to have PFE maintain backup copies of files in the same directories as the files themselves. The exact steps taken are these:

1 If the file being saved exists already, PFE renames it. The new file name is the same as the original, with the file type changed to be "**.$$$**". Any file of this name that exists already will be overwritten.

If the file *already* has a file type of "**.$$$**", no backup is taken.

2 The data in the current window is then saved to disk

Then, if you need to revert to the previous version of the file, you will find it in the same directory as the original, with a file type of "**.$$$**"

You can choose to use a different file extension to indicate a backup file with the **Options Preferences** command.

Alternatively, you can set PFE to maintain backup copies of files in subdirectories. The exact actions taken are these:

1 If it does not already exist, PFE will create a backup sub-directory in the same directory that contains the file to be backed up. The directory will be called "**$PFEBK**" by default; but if you wish it to have another name you can set this with the **Options Preferences** command.

2 The file that would be overwritten is moved into this backup sub-directory, replacing any file of the same name that is already there.

3 The data in the current window is then saved to disk

Then, if you need to revert to the previous version of the file, you will find it in the backup sub-directory, with the same name as the original file.

If the file you're editing is actually in a backup sub-directory already, PFE will warn you of this before doing anything else. You will be given the choice of creating a further backup sub-directory as described above; or of saving the file without taking a backup; or of cancelling the operation.

You do need to take care that the process of taking a backup does not require the use of any pathname that exceeds the maximum supported by the operating system..

Each backup mode has advantages and disadvantages.

- Backing up to files with a type of "**.$$$**" (or whatever your choice is) avoids creating lots of

directories, but can clutter your source directory.

- If you edit files with similar names (such as **main.c** and **main.h**) the backup for *each* will be in **main.$$$**, so you will be able to revert only the file that was saved last. It's also not obvious which file is actually backed up.

- Backing up to subdirectories guarantees that each file is backed up to a unique file, and it's easy to associate a backup with an original. However, you can end up with a lot of subdirectories.

- Backing up to subdirectories keeps all your backup files in a single place, so you can delete them in one operation.

If you don't want to keep backup copies, you have several ways of telling PFE not to do so

- You can use the **<u>Options Preferences</u>** command to disable backups completely for all files

- If you're using the **<u>File Save As</u>** or **<u>File Write</u>** commands, you can check or clear the **Keep Backup** checkbox to activate or deactivate backup for this file only, on this write operation only.

- You can use the **<u>Options Current Modes</u>** command to turn off backup for the file shown in the active window for all future write operations.

- You can use the **<u>Options Default Modes</u>** command to turn off backup for files of given types, or for all files, for all future write operations.

Of course, whichever mode you select, maintaining backup copies of all the files that you edit can consume a large amount of disk space. If you find that you are unable to save a large file, you can try to perform the operation with backup disabled.

Closing A File

When you've finished editing a file, you can close it with the **File Close** command. PFE will close all the windows that are showing the same file as the current window.

If you haven't yet saved some changes to the file, PFE will give you the opportunity to either write the file to disk before it is closed, or to abandon the operation.

If you want to close *all* the files that you are working on, use the **File Close All** command.

Closing the last, or only, window showing a file has the same effect as the **File Close** command.

If you're using PFE as an edit server for some other application, you may find it convenient to have the main window become an icon when you've closed all the files you've been working with. You can select this with the **Options Preferences** command.

Mailing A File

If you have a MAPI-compliant mail system, such as Microsoft Mail 3.0, installed on your system, you can mail files to other people directly from PFE.

To mail the file shown in the current window, simply select the **File Mail** command. PFE writes the data to a temporary file, and calls the mail system to send it as an attachment in a mail message.

If you aren't currently running the mail application, or if you aren't logged in to your mailbox, you'll see a dialog that asks you for your mail username and password. You'll then see the standard mail composition window, with the file you're sending shown in the message as the PFE icon, and you can add text or other attachments as you want.

If you're editing a file that already exists on your disk, this copy is *not altered* by this process.

Note that you'll need to have sufficient room on your disk for PFE to save the entire contents of the current window as a temporary file. This temporary file will be deleted once the mail process is complete.

Printing A File

You can print the file that is being shown in an edit window by using the **File Print** menu command. PFE will show you a dialog that lets you choose what part of the file you want to print: you can select to print either all of it, or only the part that you've selected, or the part between two line numbers.

You can specify the last line of the file as the word **end** in the rightmost of the line range edit controls.

If you have highlighted some text in the edit window, the default action in the dialog will be to print the selected text only; otherwise, the default action is to print the whole of the file.

The dialog box shows you the name of the printer that PFE is currently using - if you haven't specified otherwise, this will be your system default printer.

The file is printed using the same tab width that is being used in the window, and with line numbers if you have line numbering turned on. You can configure both these settings with **Options Current Modes** menu command.

You can quickly turn line numbering on or off by clicking the line number toggle button on the tool bar.

Setting Up The Printer
Selecting A Printer Font

Setting Up The Printer

To configure the printer that PFE is to use, use the **File Print Setup** menu command or click the **Setup** button in the dialog started by the **File Print** menu command. This will start the printer setup dialog.

The dialog lets you specify:

- The printer you wish to use

- What page margins are to be applied when you print on this device

- What font is to be used when you print to this device

You will also be able to run the printer's own setup dialog to set device-specific information.

All the settings that you make in this dialog are remembered by PFE on a per-printer basis, so you can have different settings for different printers.

The Printer Setup Dialog

Selecting A Printer Font

Clicking the **Fonts** button in the dialog started by the **File Print Setup** command starts a sub-dialog that lets you select the font to be used with the selected printer device.

Because PFE is a text editor rather than a word processor, you're restricted to choosing fixed pitch fonts, where the characters are all the same width.

PFE records the values you set in this dialog with the selected printer: the next time you choose this device, it will use the same settings.

The Printer Font Dialog

Working With Unix Format Files

Under MS-DOS, files are normally stored with the end of every line indicated by a **Carriage-Return** (CR) byte followed by a **Line-Feed** byte (LF). Most MS-DOS and Windows programs expect files to be in this format, and will not work correctly otherwise.

However, you may wish sometimes to handle files in Unix format, where lines are terminated by a single LF byte. You may have downloaded a file from a Unix system that needs conversion, or you may be using PFE to edit files on an NFS-mounted Unix file system, for example.

PFE is able to handle files in either MS-DOS or Unix format (or indeed, files using a combination of formats). By default PFE will detect that a file is in Unix format, and automatically set that format for when the file is saved.

The format that PFE will use to save a file in is shown in the status bar. You can quickly change the format by double-clicking the left mouse button in the appropriate area.

Opening Unix Format Files
Saving Files In Unix Format
Detecting Unix Format Automatically

Opening Unix Format Files

PFE handles opening and reading files in Unix format automatically, and you need take no special action to tell it that the file you are opening or inserting is a Unix file.

Depending on how have you have configured the optional file modes associated with the *file type* of the file being opened, PFE may set the **Save in UNIX format** <u>mode</u> on the file, so that when you write it to disk, lines will be terminated by single line-feed characters.

By default, files that are detected as being in Unix format when they are opened will be saved in Unix format.

Saving Files In Unix Format

PFE is able to write files to disk in either the normal MS-DOS format, using CR-LF as a line terminator, or in Unix format, using LF.

The format used to write a file is determined by one of the <u>file modes</u> - the **Save in Unix Format** mode - that you can associate with a file. You can set this mode in several ways:

- You can check the **Unix format** box in the dialog shown when you use the **<u>File Save As</u>** command. This will save the current file in UNIX format, and will also set the **Save in Unix Format** mode so that this becomes the default format when you write the file to disk again

- You can double click the left mouse button in the area of the <u>status bar</u> that shows the file format until it shows the word "**Unix**".

- You can check the **Save in Unix Format** box in the dialog started by the **<u>Options Current Modes</u>** command. This also makes Unix format the default for every time you write the file to disk

- You can have PFE set the mode automatically when the file is opened by using the **<u>Options Default Modes</u>** command. This lets you set the mode on either every file that you open, or only on those with specified file types

Detecting Unix Format Automatically

When opening a file, PFE looks at what characters are used to terminate a line. If the first line is terminated by a single Line-Feed character, PFE assumes that the file is in Unix format; otherwise, it will assume the file is in MS-DOS format.

Normally, if a file appear to be in Unix format when opened, PFE will automatically set the **Save in Unix Format** file mode, so that when you save the file it will be written in the same Unix format.

If you work with Unix files on your system, this will probably be what you want to happen. However, if you normally want to save files in DOS format, however they look originally, you can disable this feature with the **Options Preferences** command.

Typing Text

You can type text with PFE in the same way as for any other Windows application.

Normally, what you type appears on the screen, displacing other characters to the right if you're not at the end of a line. This is termed **Insert** mode.

If you prefer, you can switch to working in **Overwrite** mode, where a character you type will *replace* the one under the caret.

The mode is shown as either "**INS**" or "**OVR**" in the status bar. You can also tell by the shape of the caret: in insert mode it's a vertical line, and in overwrite mode it's a block.

You can switch between the two modes in several ways:

- Pressing the **Ins** key changes from one mode to the other

- Double clicking the left mouse button in the area of the status bar that shows the mode changes from one to the other

- You can change between the two modes, and also change other settings for the current window, with the **Options Current Modes** command

- You can set the default mode to use for files of specific types with the **Options Default Modes** command.

Remember that insert and overwrite modes apply to individual *windows*. You can have, for example, one window that shows a file in insert mode, and another showing the same file in insert mode if you wish.

Automatic Text Wrapping

Deleting Text

Pressing the **Delete** or **Del** keys will delete the character to the left of the caret, and move the caret left.

If you want delete an entire line or the last part of a line, there are two useful commands that let you do it in a single operation. The **Edit Delete Line** command deletes the *entire* line that the caret is in; the **Edit Delete To End Of Line** command deletes everything from the current position of the caret to the end of the line that contains it.

To delete a large area of text, you can <u>select</u> it with the keyboard or mouse and then press **Delete** or **Del**, or use the **Edit Cut** command.

Selecting Text

Like other Windows applications, PFE allows you to *select* an area of text and operate on it. The selected area appears highlighted.

You can select text using either a mouse, or with the keyboard. Also, the **Edit Goto Line** and **Edit Find** commands allow you to specify that all the text from where the caret starts to where it ends up is selected.

Selecting An Entire File
Selecting A Single Word
Selecting Using The Mouse
Selecting Using The Keyboard
Cancelling A Selection

Selecting An Entire File

To select the whole of the file shown in the current window, use the **Edit Select All** command. PFE will highlight all the text in the file, and move the caret to after the last character.

Selecting A Single Word

PFE gives you a quick way to select a single <u>word</u> in your text.

- With a mouse, place the cursor anywhere within the word and double click the left mouse button

- Alternatively, or if you don't have a mouse, move the caret to anywhere within the word and use the **<u>Edit Select Word</u>** command

Selecting Using The Mouse

To select some text with the mouse, hold down the left button and drag the cursor. PFE will scroll the window in the appropriate direction when you move outside the window area. All the text that you select will be shown highlighted.

To select a single word, place the mouse pointer anywhere within it and double-click the left mouse button.

You can quickly select a large area by holding down the **Shift** key as you click the left mouse button. PFE will select all the text from the initial position of the caret to where the mouse cursor is. Since PFE does not move the caret within the file as you scroll with the up and down arrows in the vertical scroll bar, this can be a convenient way to select large areas.

If you have already selected an area, holding the **Shift** key down while you press the left mouse button *extends* the selection to the position of the mouse cursor.

Remember that, although you can have an area of selected text in each of the *files* you have open, only *one* window from any particular file can have a selection.

Selecting Using The Keyboard

As with other Windows applications, PFE allows you to select text using the keyboard.

Holding down the **Shift** key while using the **cursor keys** will move the caret up and down in the current window, selecting all the text that you pass over. Normally the caret moves one character or line at a time; holding the **Ctrl** key down at the same time as the left or right cursor key will move the caret by one word at a time.

Holding down the **Shift** key while pressing the **PgUp** or **PgDn** keys will scroll the window by one window's worth of lines, selecting all the lines that are passed over. Holding the **Ctrl** key down at the same time will move you to the start or end of the file respectively.

Holding down the **Shift** key while pressing the **Home** and **End** keys will move the caret to the start or end of the current line respectively, selecting all the text passed over.

To select a complete <u>word</u>, move the caret to anywhere within it and use the **<u>Edit Select Word</u>** command.

Cancelling A Selection

To cancel a selected area of text and return it to unselected state, you should either click the left mouse button once anywhere in the window concerned; or press the **5** key on the numeric keyboard with Num Lock off.

Cutting, Copying and Pasting Text

PFE uses the standard Windows *clipboard* for cutting, copying and pasting text. You can move text around in one edit window; move it from one window to another; or transfer text to and from other applications.

You can <u>select</u> the text with the mouse or keyboard, and then operate on it with menu commands. An alternative, very quick way of moving and copying text, is to use PFE's <u>drag and drop</u> facility.

You can work with very large amounts of data when cutting, copying and pasting. If you are using the 16-bit edition on an 80286 based system, the maximum amount of data that can be handled is 80 bytes less than 1 megabyte. On an 80386 or higher, the maximum size is 16 megabytes less 64 kilobytes. With the 32-bit Edition there is effectively no limit.

However, you should normally work with small amounts of data, even if it means moving a large amount of text around in several operations instead of one. Clipboard data is stored by Windows in your system's main memory, and obviously the larger the amount you try to store, the more likely you are to run out of space. Also, because of the way Intel processors operate under Windows 3, PFE handles cutting, copying and pasting of more than 64 kilobytes of data significantly slower than smaller blocks.

Remember too that the clipboard is *not* emptied after you paste it into a file. This means that, when you exit PFE, you may still have a large amount of data stored in the clipboard taking up main memory space, which could cause you memory starvation problems when running other applications. You can reduce the amount of data stored in the clipboard by copying a single character to it before exiting from PFE; or you can use the Windows **Clipboard Viewer** application to empty the clipboard at any time.

> **<u>Cutting Text To The Clipboard</u>**
> **<u>Copying Text To The Clipboard</u>**
> **<u>Pasting Text From The Clipboard</u>**
> **<u>Dragging And Dropping Text</u>**

Cutting Text To The Clipboard

To cut some text from the current window and place it on the clipboard, you should first <u>select</u> it, then use the **<u>Edit Cut</u>** command.

PFE will delete the selected text from the edit window after making a copy of it in the Windows clipboard. Anything that was previously held in the clipboard will be lost.

Copying Text To The Clipboard

To copy some text from the current window and place it on the clipboard, you should first <u>select</u> it, then use the **<u>Edit Copy</u>**   command.

PFE will duplicate the selected text in the clipboard, and leave the window unchanged.

By default, PFE leaves the selected text highlighted after this operation. If you prefer to have the highlighting removed, use the **<u>Options Preferences</u>** command.

Pasting Text From The Clipboard

When the Windows clipboard contains some ASCII text (either put there from PFE or some other application) you can paste it into the file you're editing at the position of the caret with the **Edit Paste** command.

If you have selected any text prior to using this command, PFE will delete it before performing the paste operation.

Dragging And Dropping Text

If you have a mouse, a very quick way to move text around within a window is to use PFE's *drag-and-drop* facility, which lets you simply pick up text from one place and drop it where you want it.

To use drag-and-drop, you should do this:

1 Select the text you want to move or copy to another part of the window

2 Place the mouse cursor within the selected area. The cursor shape changes from an I-beam to an arrow to show that you can use drag and drop.

Press the left mouse button and hold it down. The cursor changes to the *drag-and-drop* cursor. If you want to *copy* the selected text rather than move it, you should also press and hold down the **Ctrl** key

3 Move the cursor to where you want to move or copy the text; the caret shows you the exact place where the text will go. Moving the mouse cursor outside the window will cause it scroll in the appropriate direction

4 When the caret is at the required location, release the left mouse button (if you're copying the text, make sure you hold the **Ctrl** key down until *after* you release the button).

If you decide that you didn't really want to carry out the operation once you've started it, release the **Ctrl** key if you're holding it down, and press the **ESCAPE** key. The mouse cursor will return to its normal shape, and you can now release the mouse button without changing the file.

Note that the selected text is *not* copied to the clipboard when you perform a drag-and-drop; anything you have in the clipboard already will be unaffected.

Normally, PFE will *move* the selected text unless you hold the **Ctrl** key down when releasing the mouse button. If you prefer, you can change this so that text is *copied* by default, and *moved* only if the **Ctrl** key is down. To do this, use the **Options Preferences** command.

Depending on how responsive your system is, and how you like to work with an editor, you may prefer to have the drag and drop facility turned off. You can do this also with the **Options Preferences** command. If drag and drop is not available, the mouse cursor does not change shape when it is over highlighted text.

Finding Text

To search for an occurrence of some text in a window, place the caret at the point where you want the search to start, and use the **Edit Find** command. This starts a standard Windows <u>dialog</u> that lets you specify what you want to look for, and other details.

The dialog box remains visible after you've made the search, enabling you to repeat it in the same or in another edit window.

<u>The Edit Find Dialog</u>

<u>Finding Special Characters</u>
<u>Repeating A Find Operation</u>

Finding Special Characters

When you're searching you may want to look for occurrences of characters that you can't type in the edit control of the Find dialog. PFE lets you specify them using a special notation:

\f	represents a Form Feed character
\n	represents the end of a line
\t	represents a TAB character
****	represents a single '\' character

To specify an arbitrary character code you can use the notation "**\xnn**", where "**nn**" is two hexadecimal digits. The null value "**\x00**" is not permitted.

Repeating A Find Operation

There are two ways of repeating a text search. Because the <u>Find dialog</u> remains visible until you explicitly dismiss it, you can repeat a search - either exactly or with changes - simply by pressing the **OK** button. You can perform the search in any window.

However, on standard VGA screens especially, the dialog can take a large amount of space, so you may prefer to dismiss it and use the **<u>Edit Repeat Last Find</u>** command to repeat a search. This command repeats the last search operation you did *exactly.*

Replacing Text

To search for an occurrence of some text in a window, and replace it with some different text, place the caret at the point where you want the search to start, and use the **Edit Replace** command. This starts a standard Windows <u>dialog</u> that lets you specify what you want to look for, what to replace it with, and other details.

The dialog box remains visible after you've performed the operation, so you can repeat the operation in the same or in another edit window.

<u>The Edit Replace Dialog</u>

<u>Replacing Special Characters</u>
<u>Repeating A Replace Operation</u>

Replacing Special Characters

When you're replacing text you may want to look for or insert characters that you can't type in the edit control of the Replace dialog. PFE lets you specify them using a special notation:

\f	represents a Form Feed character
\n	represents the end of a line
\t	represents a TAB character
****	represents a single '\' character

To specify an arbitrary character code you can use the notation "**\xnn**", where "**nn**" is two hexadecimal digits. The null value "**\x00**" is not permitted.

Repeating A Replace Operation

There are two ways of repeating a text replacement operation. Because the Replace dialog remains visible until you explicitly dismiss it, you can repeat an operation - either exactly or with changes - simply by pressing one of the **Find Next**, **Replace** or **Replace All** buttons. You can perform the operation in any window.

However, on standard VGA screens especially, the dialog can take a large amount of space, so you may prefer to dismiss it and use the **Edit Repeat Last Replace** command to repeat the process. This command repeats the last replace operation you did *exactly.*

Text Processing

Although PFE is not intended to be used as a word processor, it does support some text processing features that are of use to those editing program source, and a few of general applicability.

Automatic Text Wrapping
Indenting And Undenting Text
Changing Case
Inserting Characters by ASCII Code
Handling Braces

Automatic Text Wrapping

When you're typing ordinary text, rather than a program source, it's often convenient to have the program you're using handle the business of fitting the text into the available line width.

The quickest way to activate text wrapping for a window is to double-click the left mouse button in the wrap area of the status bar. This will show the text "**No Wrap**" when wrapping is not active. The double click action will turn wrapping on, using the currently set wrap column, which by default is column 72.

You can also switch wrapping on and off, and also define the wrap column, for the current window, using the **Options Current   Modes** command.

You can also cause text wrapping to be activated automatically for all files of a specific type by using the **Options Default Modes** command.

Whether wrapping is active, and the wrap column used, is set on a per-window basis. You can, if you want, edit the same file in two different windows with different wrap settings.

In any window for which wrapping is active, PFE monitors as you type, and whenever you type a non-space character at a screen position to the right of the defined wrap column, it will automatically break the line you're typing in at a suitable place, moving some text and the insertion point down to the next line.

The points at which PFE will break a line vary with the *language type* defined for the window with the **Options Current   Modes** command.

Language type "(none)" or "C"

The line will be broken at the closest white space character or hyphen to the defined wrap column.

Language type "TeX"

The line will be broken at the closest white space character to the defined wrap column

Note that PFE will wrap the line you're typing *only* when you're typing at the right hand end of it. If you move the caret to somewhere within the line and type, the line won't be wrapped even if it extends past the wrap column.

PFE does not perform any sort of text justification.

Indenting And Undenting Text

When you're editing program source, you'll very often want to change the indentation of some or all of it. PFE gives you a powerful facility to change the indentation level of either a single line, or of a larger block of text.

To indent a *single* line, place the caret anywhere within it and use the **Edit Text Indent** command. PFE will move the entire text of the line to the right by one tab stop, inserting either a single tab character or a number of spaces, as appropriate for the window. Lines that are empty are not affected.

To undent a *single* line, place the caret within it and use the **Edit Text Undent** command. PFE will move the text left by one tab stop: if the line starts with a tab character, this will be deleted; otherwise, PFE removes as many space characters as it can up to the width of a tab stop. A line that does not start with either a tab character or a space will not be altered.

If you want to indent or undent more than one line, the procedure is similar; but now you indicate the lines you want to change by selecting them. Now when you use either of the commands, PFE will alter all the lines that are selected. Unlike most operations that affect selected text, indenting and undenting operates on the whole of any line containing a highlight, and not just on the highlighted text itself.

Changing Case

PFE lets you quickly change the case of the text that you've typed. First, you should <u>select</u> the text whose case you want to change.

The **<u>Edit Text Uppercase Selection</u>** command will make all the characters you've selected upper case; the **<u>Edit Text Lowercase Selection</u>** command makes them all lower case.

Inserting Characters by ASCII Code

PFE allows you to insert characters with arbitrary ASCII codes between 1 and 255 that you cannot type on the keyboard. This facility permits you to insert control characters such as **ESC** and **Form Feed**, or characters with codes greater than 127.

The **Edit Text Insert ASCII Code** command starts a <u>dialog</u> that allows you to specify the ASCII code to insert.

Handling Braces

PFE provides several commands that allow you to manipulate *brace characters* such as brackets and parentheses, which often delimit sections of text or of program sources.

The definition of a brace character varies, depending on what language type you have set in a window's window modes. If you have defined the language type to be "**C**", braces are defined as the characters that have that syntactic meaning in the language - opening braces are **{ [** or **(**, and closing braces are **}]** or **)**.

If you have set the language type to be "**none**", brace characters are defined as those commonly used in text or many other languages - opening braces are **{ [(** or **<**, and closing braces are **}])** or **>**.

Two commands allow you easily to locate braces that match each other. If the caret is positioned on either an opening or a closing brace character, the **Edit Text Match Brace** and **Edit Text Match Brace Select** commands will move the caret to the matching one, having regard for nesting of braces. The latter command will also highlight all the text between and including the brace characters.

The **Edit Text Widen Brace Select Command** command helps you locate and manipulate brace-delimited sections of text. If the caret is positioned anywhere between two matching braces, the command will move the caret to the opening brace character and select the text between and including them. The next use of the command will widen the selection to include the brace characters that completely enclose this selected area, and so on. The result is that, in C language source, for example, you can select larger and larger syntactic elements with simple repeated keystrokes, and copy, delete or otherwise manipulate them as you choose.

In locating matching brace commands, PFE's actions are again moderated by the defined language type. For a language type of "**none**", all characters are examined, and there is no syntactic meaning to any text.

For a language type of "**C**", however, the search is made with regard to C language syntax. Brace characters within comments, delimited strings or pre-processor directives are disregarded, as are those defined as character literals. In most cases, PFE will be able accurately to match a brace character. However, some complex syntax may mislead it, especially if the syntax is invalid or incomplete.

Undoing Edit Actions

PFE includes a powerful and configurable system that will let you correct your editing mistakes easily. The edit changes you make are recorded in memory, and you can reverse them with the **Edit Undo** command, which can be initiated quickly by clicking on a **tool bar** button.

When recording the edit changes you make, PFE handles them as a sequence of *actions*. By default it will record the last 32 actions performed on each open file; but you can change this to any number between 8 and 128 with the **Options Preferences** command.

In most cases, it is obvious what an undoable *action* is. For example, deleting a line with the **Edit Delete Line** command is a single action, as is pasting from the clipboard with **Edit Paste** or inserting a template with **Template Insert**.

However, typing and deleting single characters is handled somewhat differently. PFE does not regard each keypress as an *action*: rather, it stores as one action all the keystrokes you make, until you begin doing something else. Undoing the action then reverses the whole collection of keystrokes, rather than just the last one.

For example, simply typing characters accumulates keystrokes in a *typing* action; deleting with the **BackSpace** key accumulates keystrokes in a *delete backwards* action, and deleting forwards with the **Delete** key accumulates them in a *delete forwards* action. PFE only stops accumulating keystrokes in an action when you do something *different*:. For example, PFE would stop accumulating keystrokes in a *typing* action if you began to delete characters; and it would stop accumulating keystrokes in a *delete backwards* action if you began typing characters again.

Accumulation of keystrokes will also stop if you use any key that moves the caret; move the caret with the mouse; or perform any command.

Obviously, recording the edit changes you have made to a file consumes memory; and if you have specified that PFE should record a large number of edit actions, and cut and paste large amounts of data, you may use up very large amounts indeed.

PFE provides several techniques to keep memory usage under control:

- The **Edit Clear Undo Actions** command lets you discard all the recorded actions for the current file, freeing off the memory used, at any time. Of course, once you have used this command, you can no longer reverse any changes.

- When you write a file to disk, PFE will by default clear the recorded actions for the current file for you, releasing the memory used. You can suppress this, and retain all recorded actions until you decide to clear them yourself, with the **Options Preferences** command.

Templates

Templates are pre-built sections of text that you can include into your files, saving you from repeatedly typing the same things.

For example, you might like to document your C procedures by preceding each one with an explanatory section like this:

```
/************************************************

    start_editor
    ------------

    This procedure starts the system in edit
    mode

************************************************/
```

Rather than type the lines of asterisks every time, you could set up a *template* containing the text:

```
/************************************************


<???>
      -------------

************************************************/
```

and insert it automatically in a simple operation. The "**<???>**" characters are a *template mark* to show where you have to type in extra items (such as the name of the procedure in this example). PFE lets you move between these   marks with a single keypress, so that editing is very simple and convenient.

Templates are held in *template files*, which you can regard as a sort of library. You can have as many template files ready for use, or *attached* as you want, and you can insert templates from any of them at will. Template files are not ordinary text files and can't be edited directly, but the **Template** menu provides you with options for creating template files, attaching and detaching them, creating, editing and deleting templates.

Each template file can hold a maximum of 64 templates. The total space taken up for data in a template file cannot exceed a total of approximately 62 kilobytes. Each line in a template requires one byte for each character, plus two extra bytes; and one further byte is required for each template.

You can attach template files manually at any time as you need them. However, you'll probably find that you create a set of templates that you use all the time: you can arrange for PFE to attach these automatically when it starts up. You can also have a set of templates that are specific to a particular project, and this also can be attached automatically.

Creating A Template File
Attaching A Template File
Creating A Template
Inserting A Template
Editing A Template
Deleting A Template
Saving A Changed Template File
Detaching A Template File

Creating A Template File

Before you can store a template you must have a *template file* to place it in. The **Template Create File** command starts a standard <u>dialog</u> that lets you specify the name of the file you want to create. PFE will create the file you specified and write a small amount of binary control information into it.

You should use the file type **.tpl** when you create a template file. You should not try to template files directly, as they do not contain normal text.

When you create a template file it is automatically <u>attached</u> and ready to use.

Attaching A Template File

Before you can access any of the templates contained in a template file, you must first *attach* it. PFE loads the file into memory and constructs an index to the templates it contains, after which you can manipulate it.

You attach a template file with the **Template Attach File** command. This starts a <u>dialog</u> that lets you select the file you want to attach.

If you wish, you can attach a template file in *read only* mode. You will be able to insert templates contained in this file into files that you are editing, but you will not be able to add new templates to it or edit existing ones within it.

Having a template file attached uses some memory: PFE uses global memory to hold the data within the template file, and some control blocks. You can recover the memory used by an unwanted template file by *detaching* it with the **Template Detach File** command.

Automatically Attaching Template Files

Automatically Attaching Template Files

If you like, you can have PFE attach some template files automatically when it starts up, rather than attaching them manually one by one with the **Template Attach File** command. You can specify the files in two ways:

- As it starts, PFE scans the current directory for a file called **auto.tpl** and attaches it if it is a valid template file

- You can specify a   template directory with the **Options Preferences** command.
  When it starts, PFE will look for a directory with the specified pathname, and will attach *all* the files it contains with a file type of **.tpl**

Creating A Template

To create a template, use the **<u>Template New</u>** command. This opens up an edit window that is exactly the same as any other; the sole difference is that the window caption will show that you're editing a template.

You can edit inside the window exactly as in any other. You have an additional facility, though: you can use the **<u>Template Insert Mark</u>** command to insert a *template mark*. This is a character sequence (actually the characters **<???>**) that are automatically searched for by the **<u>Template Find Mark</u>** command. You can use template marks to hold the position for things to be filled in once you've inserted a template into your text file.

In order to use the template once you've finished editing it, you must store it in a template file. You can create a template at any time, but to store it you must have a template file attached. **<u>Template Store As</u>** command starts a dialog that shows you all the template files that you've attached. You should select the file you want to store the template in, then type the template's name. This can be any sequence of up to 16 characters, used to identify this particular template.

PFE will write the template information into its memory copy of the template file, and you can then insert the template into files at will. However, the template will *not* be saved to disk until you use the **<u>Template Save File</u>** command.

Inserting A Template

At any time when you're editing, you can insert a template from any attached template file at the position of the caret.

The **Template Insert** command shows you a <u>dialog</u> that gives you a list of all the attached template files. When you select one of the files you'll see a list of all the templates that it contains; you can then select the particular template that you want to insert.

PFE will read the template data from the template file and insert it into the current window at the position of the caret. Unlike the case when you're inserting another file or pasting from the clipboard, the caret remains at the *start* of the inserted data. This allows you to use the **Template Find Mark** command to move automatically to any template marks that you've put in the template to show where extra text needs to be inserted.

Editing A Template

In order to change a template that you've stored in a template file you need first to <u>attach</u> the file, then use the **<u>Template Edit</u>** command. Since template files do not contain plain text, you shouldn't attempt the edit them directly.

The command shows you a <u>dialog</u> that lists all the attached template files. Selecting one of the files shows you a list of all the templates it contains, and then you can select the template you want to edit.

When you've selected a template, PFE will place it in an edit window for you. The window is exactly the same as one for editing a normal text file, but the caption will show you that it's actually a template you're editing. You can edit the template in any way you like.

Once you've finished your edit, you can update the template file with the **<u>Template Store</u>** command. This will delete the old version of the template from the template file and replace it with the new one. If you want to save the contents of the edit window as a template with a different name, or into a different template file, use the **<u>Template Store As</u>** command.

A template *file* is *not* updated on disk when you finish editing a template. To preserve your changes you need to explicitly save it using the **<u>Template Save File</u>** command.

Because a template is not regarded as an ordinary file, you can't use the **<u>File Save</u>** or **<u>File Save As</u>** commands. If you need to save the template into an ordinary file, use the **<u>Edit Copy</u>** command to copy the data to the clipboard and paste it into a new window.

When you're editing a template, you can use the **<u>Template Insert Mark</u>** command to insert a *template mark*. This is a character sequence (actually the characters **<???>**) that are automatically searched for by the **<u>Template Find Mark</u>** command. You can use template marks to hold the position for things to be filled in once you've inserted a template into your text file.

Deleting A Template

To delete an unwanted template from a template file, you must first <u>attach</u> it, then use the **Template Delete** command.

This starts a <u>dialog</u> that allows you to specify one or more templates in a template file and delete them. The template *file* is *not* updated on disk when you delete a template: to preserve your changes you need to explicitly save the file using the **Template Save File** command.

Saving A Changed Template File

When you edit a template, create a new one, or delete one, PFE makes the changes only to the copy of the template data that it keeps in memory. The template file on disk is not altered in any way.

To make your changes permanent, you need to use the **Template Save File** command This starts a dialog that shows you the names of all the template files that have been altered, allowing you to select those you want to write back to your disk.

Detaching A Template File

If you don't want to use the templates contained in a template file any more, you can regain some memory by removing the in-memory copy of the file with the **Template Detach File** command.

The command will start a dialog that lets you choose the template files to detach. Once you've detached a file, you won't be able to use the templates it contains until you attach it again.

PFE will not let you detach a template file if you're currently editing any of the templates it contains.

File And Window Modes

PFE allows you to change many aspects of how it operates in any particular window by setting various *modes*. Most mode values are simply *yes-or-no* settings, such as "*are trailing spaces to be removed?*" and "i*s the file read-only?*". Others take specific settings, such as the width of a tab stop or a colour selection.

Some modes, referred to as *window modes*, apply only to one specific window. If you're editing the same file in several windows at the same time, you can apply different window modes to each window.

File modes, by contrast, apply to a file in *all* the windows that you are using to edit it. Whichever window is the current one when you change a file mode, the value will be applied to all the others.

You can set mode values in two distinct ways.

Setting current mode values

At any time, you can use the **Options Current Modes** command to set the modes that should apply to the current window, and the file that you are editing within it. You can use this method to set some specific values to help you in a particular editing task. These mode values are not stored.

Setting default mode values

Whenever PFE creates a window, for instance after you've used a **File Open** command to open an existing file, it will set the initial mode values to a default set. The **Options Default Modes** command lets you tailor these initial values to suit your preferences.

In fact, you can set many *different* sets of default mode values, which are used in the appropriate circumstances. You can set defaults to be used:

- When PFE creates a window to edit a template

- When PFE creates a window to show the output of a DOS command captured by the **Execute DOS Command with Capture** command

- When PFE creates an empty edit window with the **File New** command

- When you open an existing file whose file name (less the directory path) matches an entry in one of your mode groups. . You can define up to 32 sets of such default values, and associate each with up to 16 wildcarded filename patterns.

- When you open an existing file whose name is not matched in any of the mode groups referred to above

Initially, PFE uses the same set of default values for all these cases, but you can configure each case as you wish independently of the others

Setting Modes For A Window Or File

If you want to change the Window Modes for the current window, or the File Modes for the file that's being shown in it, use the **Options Current Modes** command, which starts a <u>dialog</u> that lets you alter all the settings.

The dialog box reminds you of the name of the current file at the top. You should set the modes that you want by checking or unchecking the boxes, then press **OK** to action them.

A *mode group* is a collection of file name patterns (such as **"*.c"**, **"readme.*"** and **"*.lsp"**) and a set of corresponding <u>file and window modes</u>. The file patterns can contain the normal DOS wildcard characters **"*"** and **"?"**

When PFE opens a file, it searches the defined mode groups to see if it can match the file name in one; if it does, it applies the corresponding mode values. Note that the directory path parts of filenames are ignored in this search.

You can edit the file patterns that make up your mode groups, and the modes that correspond, with the **<u>Options Default Modes</u>** command

Moving To Specific Lines

You have several ways of moving to a specific line in a file. You can use the normal movement keys and watch the file position in the status bar at the bottom of the main window, or you can turn on file line numbers and use those to guide you.

Alternatively, you can use the **Edit Goto Line** menu command. This starts a dialog that lets you enter the desired position directly; you can move to a specific line number, or to a specific character within a specific line.

Window Manipulation

PFE uses the standard Windows Multiple Document Interface to allow you to edit as many files as you wish simultaneously. You can use all the normal MDI techniques to manipulate the windows, and there are some further facilities specific to PFE.

Selecting A Window
Moving And Sizing A Window
Duplicating A Window
Closing A window
Widening A Window
Tiling Windows

Selecting A Window

You have several ways of moving from one edit window to another.

The **Window Next** command moves you between windows that are not currently icons - any iconic windows are not restored by this.

With a mouse, you can click on any window to activate it. To prevent you accidentally moving the caret when you activate a window with the mouse cursor within the edit area, PFE will not change the caret position until you click a second time.

The **Window** menu contains one item for each of the first 9 edit windows; clicking the item that identifies the window will activate it (if the window is an icon, it will be restored). If you have more than 9 windows open, you'll see a **More Windows** item, which will run a dialog that will list all the existing windows.

Closing A Window

To close a window, you can use select the **Close** item from its system menu, double-click in the system menu box, or use the **Window Close** command.

If the window is the only one currently showing a file, PFE will act as if you'd used the **File Close** command and close the file also; if you have made any changes to the file you'll be prompted to see if you want to save them, discard them, or cancel the operation.

Moving And Sizing A Window

You can iconize any edit window by clicking on its minimize box, or via its system menu. You can also iconize *all* the edit windows that exist in one operation with the **<u>Window Iconize All</u>** command.

You can maximize a window so that it fills the full extent of the main window by clicking its maximize box, or via its system menu. You can also double-click anywhere on the window's caption bar to do this.

You can move a window around or resize it with the mouse, or via the window's system menu.

Duplicating A Window

PFE allows you to have as many windows as you want showing the *same* file - for example, you might find it convenient to have one window showing the start of a file, and another showing the end of it. You can also apply different window modes to the windows for various effects.

The **Window Duplicate** command will make an *exact* duplicate of the current window. PFE will create a second window that is the same size as the original, with all the same window modes, showing the same text.

You can edit the file concerned in any of the windows that are showing it, but there will be a small performance penalty, as PFE needs to synchronise a change made in one window with all the others.

Although you can select text freely when you have multiple windows on a file, only one of the windows can have an active selection at any time. If you select some text and then activate another window on the same file, the selection will be cleared.

Widening A Window

Often you may want to make a window as wide as possible, so that you can see the maximum possible amount of text in each file line. You can achieve this in the normal way that Windows allows you to resize a window; but an easier way is to use the **Window Widen** command.

PFE will automatically make the current window as wide as possible, moving it to the far left of the main window. The window's depth and vertical position are not affected.

Tiling Windows

PFE allows you automatically to position the windows you're working with to make best use of the available screen area. You can tile windows either *vertically* or *horizontally*.

Horizontal Tiling

The **Window Tile Horizontal** command arranges your windows in columns, so that you can see as many lines in each as possible.

Vertical Tiling

The **Window Tile Vertical** command arranges your windows one above the other on the screen, so that you can see as much of each line in as many windows as possible. Depending on the size of your display, only 2 or 3 windows can be arranged in this way without becoming too small to use; excess windows are tiled in a horizontal pattern in the lower part of the screen.

Windows holding the output from commands run by the **Execute DOS Command To Window** command are preferentially arranged at the top of the screen.

In each of the tiling cases, PFE will arrange any window icons neatly towards the bottom of the screen, and will always keep the bottom row of icons visible. When tiling vertically this can reduce the usable size of the screen by too much, especially if you have more than three non-iconic windows.

You can tell PFE to use the *full* depth of the screen for vertical tiling with the **Options Preferences** command.

Screen Fonts

By default, PFE will use the fixed pitch **System** font to display text in edit windows. However, it will allow you to use *any* fixed pitch font that exists on your system, as well as the standard ANSI and OEM fonts.

To select the screen font that you want PFE to use in you edit windows, use the **<u>Options Preferences</u>** command.

Selecting a screen font has no effect on any fonts you may have selected for printers.

The Status Bar

PFE normally shows you a **status bar** at the bottom of its window. This gives you several items of useful information, and also gives you a quick way of changing some settings and executing a few commands.

The status bar shows you:

- The line number and column number where the caret is. Double clicking the left mouse button in this area executes an **Edit Goto Line** command.

- The total number of lines in the file.

- Whether the file has changed : you will see a **#** if it has.

- Whether the file is *read only* : the characters **RO** will show if it is, and **WR** will show if you can alter it. Double clicking the left mouse button in this area will change between the two settings.

- The command sequence prefix key - that is, whether you've typed an active prefix key and PFE is waiting for you to supply the second key of the sequence.

- Whether your keystrokes and menu selections are being recorded in a keyboard macro. Double clicking the left mouse button in this area will turn recording on and off.

- Whether PFE will automatically wrap lines that you type, and the column number past which this wrapping will take place. Double clicking the left mouse button in this area will turn wrapping on and off.

- Whether the file will be saved in DOS format or Unix format. Double clicking the left mouse button in this area will change between the two formats. You cannot change the save format of a file that is marked as *read only*.

- Whether typing is to be *inserted* into what's there already, or will *overwrite* it : you see **INS** or **OVR** respectively. Double clicking the left mouse button in this area will change between the two settings, as will pressing the **Ins** key.

- The state of the **Num Lock** and **Shift Lock** keys

You can turn off the status bar and make a little more room for showing the files you're editing, or turn it on again, with the **Options Preferences** command. On slower machines you will probably see a noticeable performance increase when you have the status bar turned off.

The Tool Bar

Besides controlling PFE from the keyboard, or by using the menu, you can also use the *tool bar* to make access to the most common operations simple.

The tool bar is a collection of coloured push buttons. By default it appears at the top of the screen below the menu, but you can <u>place</u> it at any of the main window's edges, or have it floating as a separate window.

You can invoke the function associated with a tool bar button by clicking the left mouse button when the mouse pointer is within it. Some buttons have more than one function, and you invoke these by holding either the **Shift** key, the **Ctrl** key, or both, down as you *release* the mouse button.

When you click the left mouse button on a toolbar button, PFE will show some text explaining its use in the <u>status bar</u> at the bottom of the screen. If you decide you don't want to perform the function, move the mouse pointer out of the tool bar area before releasing the mouse button.

The topic **Description of the Tool Bar Buttons** below describes each of the buttons fully. Note that in this help file, the buttons are drawn as they appear by default on screens with 1024 x 768 resolution. Some buttons may appear slightly differently on other screens.

You can also obtain help on the function of any button in the tool bar by using the **<u>Edit Screen/Menu Help</u>** command, positioning the question-and-arrow cursor on the button, and clicking the left mouse button once.

Not all of the tool bar buttons will be usable at all times: for example, the **Print** button can't work if you don't actually have a file open. The buttons that you *can* use will display icons in colour; those that aren't currently available will show grey shadow images.

> **<u>Description of the Tool Bar Buttons</u>**
> **<u>Moving the Tool Bar</u>**
> **<u>Setting The Tool Bar Size</u>**

Setting The Tool Bar Size

Normally, PFE selects the bitmaps it draws for the tool bar buttons to match the resolution of your screen, so that the tool bar appears approximately the same physical size on all screens. However, if you have a large screen with a high resolution, you may want to force PFE to use a smaller toolbar for a better appearance.

You can override the automatic sizing of the tool bar with the **Options Preferences** command. The allows you to force PFE to use a small, medium or large set of tool bar buttons.

Moving The Tool Bar

By default, the tool bar appears as a row of button at the top of the screen below the menu. You can choose to place it at any of the main window's edges, or you can have it appear as a small floating window that you can place anywhere at all in the edit window. You can also, of course, choose to turn it off completely.

One way of moving the tool bar is to use **Options Preferences** command. This allows you to *hide* the tool bar or *show* it, or specify where you want it to be placed.

If you have a mouse, you can also *drag* the tool bar to where you want it. You need to do this to start:

-   If the tool bar is at one of the window edges, place the mouse pointer within it but *not* over one of the buttons. The cursor will change to the shape of a hand. Press the left mouse button and hold it down. The cursor shape now changes to a four-pointed arrow.

-   If the tool bar is a floating window, put the mouse pointer over the caption bar at the top, press the left button and hold it down.

Now *drag* the tool bar to where you want it to be placed and release the left mouse button. If the mouse pointer is close to one of the main window's edges, the tool bar will be placed on that edge; if the mouse pointer is nearer the middle of the main window, the tool bar will turn into a floating window.

PFE will remember where you've placed the tool bar, and will put it there the next time you start the program.

Key Mapping

PFE allows you to change the details of what keys you press to perform most of its operations. For instance, you could set the **Ctrl+A** key so that pressing it executed the **Template Attach File** command if you found that convenient.

The operations that PFE performs are called *functions*, and these have fairly descriptive names such as **FileOpen** and **EditTransposeCharacters**. Most of the functions are similar in name to the commands used to control PFE over a DDE link, and to those on the menus.

The relationship between a key on the keyboard and the function that is performed by pressing it is known as a *key mapping*. You can have as many keys as you want mapped to a function.

You can store your personalised key mappings so that they are automatically applied every time you start PFE. You can also save them in key mapping files, so that different people using your system could load their own mappings; or so that you could have different mappings for different purposes

What Keys Can Be Mapped
Changing The Key Mappings
Loading Different Key Mappings

The Default Key Mappings
Dictionary Of Functions

What Keys Can Be Mapped

PFE is controlled from the keyboard using two different types of key sequences. You can map almost all the available key combinations, so you should be able to find ones that suit your way of working.

Firstly, PFE uses *single-character* key presses such as **Ctrl+A**, **Shift+BackSpace**, **Alt+X** and **F7**, in a similar way to a standard Windows application. If you wished, you could set up PFE to operate entirely with such characters; this is how it will work by default.

The second mode uses *two-characters sequences*, where you first press a *prefix key*, and then press a second key. This form is similar in approach to the **EMACS** editor or the **WordStar** word processor. By default prefix keys are not used, but you can <u>activate</u> any or all of **Esc** key and **Ctrl+A** to **Ctrl+Z** as prefixes. A prefix key can be followed by any valid single-character key, so the number of mappable key combinations you have available is vast.

PFE does not allow you to remap some of the keys that have standard functions within Windows. Thus, you can't map **Alt+Esc** or **Alt+Tab**, for example, as these are the standard way of moving between task windows.

If you map keys such as **Alt+F** or **Alt+W**, they will no longer function as menu hot keys to drop down the appropriate menus. However, you can access the menus by pressing and releasing the **Alt** key alone, and using the cursor keys to navigate to the required menu.

Changing Key Mappings

To change the mappings of keys to functions to be as you want, use the **Options Key Mapping** menu command. This starts a <u>dialog</u> from which you can choose your requirements.

> **Defining and Changing Key Mappings**
> **Changing The Active Prefix Keys**
> **Saving A Key Mapping**
> **Loading A Key Mapping**

Defining and Changing Key Mappings

To map a key that is currently unmapped to a function, select the **<u>Options Key Mapping</u>** command, which starts the key mapping <u>dialog</u>.

Saving A Key Mapping

While you are running the key mapping <u>dialog</u>, you can choose to save your new mapping at any time.

If you used the "**/k**" <u>command line option</u> to <u>load</u> a specific key map file on start-up, you will need always to use **Save As** button and name the file you want to save the changes to explicitly.

If you didn't use the "**/k**" option, you'll still be able to use the **Save As** button to write your changes to a specific file. However, you'll also be able to use the **Save** button to save the changes to the **pfe.key** file in your Windows directory, which PFE will by default load when it starts up.

Loading A Key Mapping

PFE allows you to save as many different sets of key mappings as you wish, so that different people using your system can each have their own favourite settings; or you can have different sets of your own for different purposes.

Key mappings are stored in files with type "**.key**". When it starts up, PFE will automatically try to load the key mappings from **pfe.key** (16-bit Edition) or **pfe32.key** (32-bit Edition) in your Windows directory unless you suppress this on the command line as described below. You can change the pathname of the default key mapping file with the **Options Preferences** command.

FROM THE COMMAND LINE

To load a set of key mappings other than the default when PFE starts, you can use the "**/k**" command line option to name any key map file. For example, starting PFE with a command line

```
pfe /k c:\keymaps\uemap.key
```

would load the key mappings in the file **c:\keymaps\uemap.key**. If you have a standard key map file in your Windows directory, but want to use the built-in key mappings described in this help file instead, use the "/k" option and give the filename as the single character "**-**" (minus).

Note that when you use the "**/k**" option, PFE will not save changed mappings to the file you specify, or to the standard key map file, by default. When saving you will need always to give an explicit file name.

FROM THE KEY MAPPING DIALOG

To load a set of key mappings from another file when PFE is running, select the **Options Key Mapping** command, which starts the key mapping dialog. Clicking the **Load** button will cause PFE to ask you the name of the file of mappings that you want to load.

Changing The Active Prefix Keys

By default, no prefix keys will be active when you start PFE. This means that all the control keys are available to be mapped to functions, and you will not be able to use any two-character combinations to perform actions.

To activate or de-activate prefix keys, select the **Options Key Mapping** command, which starts the key mapping dialog, and click the **Configure** button to the right of the drop-down list of prefix keys.

This starts a sub-dialog that lets you activate or deactivate keys as prefixes. You can use any or all of **Esc** or **Ctrl+A** to **Ctrl+Z** as prefixes.

Keyboard Macros

PFE is able to record key strokes and menu selections in a *keyboard macro*, which you can replay as many times as you wish to make repetitive operations easier.

The **Macro Start Recorder** command will start the recorder. PFE will record anything you type into an edit window, and any menu items that you select into the keyboard macro buffer; recording will continue until you use the **Macro Stop Recorder** command.

Once you've recorded a keyboard macro, you can replay it with the **Macro Replay** command. PFE will re-issue all the keystrokes and menu selections that have been recorded, just as if you were performing them yourself.

It's up to you to ensure that it's sensible to replay a keyboard macro. For example, if you don't have a file open, it's not possible for PFE to insert any characters. So that you don't accidentally ask PFE to perform some impossible operation, replay of the keyboard macro will be automatically halted if any command or keystroke fails. This can be very useful if you're performing searches: macro replay will stop if the target string is not found.

Although it records menu selections made by both keyboard and mouse, PFE does *not* record other mouse operations. You should, therefore, take care to use only the keyboard equivalents of mouse operations when you're recording a macro.

PFE also does not record what you type into dialog boxes. Thus, if you record an **F2** key for a search, you'll see the dialog box every time you replay the key, and will have to fill in the details again. In many cases, this is how you would want the editor to operate, but sometimes you actually want to repeat a search operation exactly every time. To achieve this, you can type **F2** to perform a search *before* you start recording; then, within the macro you can use the **Shift+F2** key to repeat the search exactly without using the dialog box.

PFE provides only *one* keyboard macro; starting a recording will automatically delete any previous macro. In a future release you'll be able to select from multiple keyboard macros, and to record them to files for use in subsequent edit sessions.

Ending a PFE Session

You can end your PFE session by selecting the **Close** item in the system menu, by double clicking in the system menu box, or by using the **File Exit** command.

Before closing, PFE will check whether you have made any edit changes that have not yet been saved to disk. If you have, it will prompt you for each unsaved file in turn, asking you if you want to save the changes or to discard them, You also have the option of cancelling the process and remaining in PFE.

Exiting Windows

When you're developing programs that may not behave well and cause damage to the system, you'll often find it necessary to exit Windows and restart it, and maybe even to reboot your machine. PFE provides you an option on its *system menu* to close Windows in a variety of ways very quickly and simply.

The facilities provided by this option vary, depending on whether you're using the 16-bit or 32-bit edition.

Exiting Windows 3.1x
Exiting Windows NT or Windows 95

Exiting Windows 3.1x

To exit from your Windows 3 system, click on the system menu box at the top left corner of the main window and select the **Exit Windows** item. This starts a <u>dialog</u> that lets you perform on of the following actions:

- Exit from Windows 3 and return to your DOS prompt

- Exit from Windows 3 and then restart it automatically

- Exit from Windows 3 and then perform a warm reboot of your system

- Exit from Windows 3, run a single DOS command automatically, then restart Windows automatically

Exiting Windows NT or Windows 95

To exit from your Windows NT system, click on the system menu box at the top left corner of the main window and select the **Exit Windows** item. This starts a <u>dialog</u> that lets you perform on of the following actions:

- Shutown all the applications that you have started and log off from the system

- Shutdown all applications in the system, and close down Windows. You will be notified when it is safe to power your computer off

- Shutdown all applications in the system, close down Windows, and automatically reboot it

Running DOS Commands

PFE gives you two ways of interacting with DOS applications.

- You can start one or more DOS sessions running your command processor, within which you can run any DOS program. These sessions run independently of PFE; if you want to edit the output of any program you run in one, you'll need to use the clipboard to paste it into an edit window

- You can start a DOS application, such as a compiler, and arrange for PFE to capture its output as it runs. When the command completes, you'll see the results in an edit window. You can run external commands, or those internal to your command processor, in this way.

<u>Starting A DOS Prompt</u>
<u>Running A Command And Capturing Output</u>

Starting A DOS Prompt

The **Execute DOS Prompt** command starts up a DOS session, in which you can issue any DOS commands you wish.

If you have defined a command processor with the **Options Preferences** command, this program will be run.

If not, PFE will look at the environment variable **COMSPEC**, and if this is defined, will run the program named in it.

If neither of the above options specifies a command processor, PFE will start **command.com** (16-bit Edition) or **cmd.exe** (32-bit Edition)

The starting directory for the DOS session will be PFE's own current directory. This can be changed through the standard file opening dialogs run by items in the **File** menu.

You can start as many DOS sessions as you wish, and you can continue to use PFE while a session is running.

DOS sessions started in this way run independently of PFE. If you wish to edit anything that appears in their windows, use the standard clipboard mechanisms to paste it into PFE.

Running A Command And Capturing Output

PFE allows you to issue a DOS command, such as a compiler driver, and capture the output into a new edit window for inspection.

To do this, use the **Execute DOS Command To Window** command. This starts a dialog that lets you specify various things about the command you want to run

The dialog allows you to set:

- The command line to be executed. This can be any legal DOS command, including built-in commands such as **dir.** If you are using the **4DOS** command processor, you can also use aliases and command sequences separated by the current delimiter character. You can also have PFE substitute parts of the name of the file showing in the current window automatically for you.

- The working directory to be used. PFE will make this the current directory for the command process before the command is issued (PFE's own current directory setting is not affected)

- Whether PFE is to minimize itself into an icon once the command has been issued

- Whether PFE is to sound a beep once the DOS command has completed.

- Whether PFE is automatically to save any altered files before running the command

- Whether PFE is to show the output in any previously used command output window, or to create a new one

- Whether the output window is to scroll automatically to show you the end of the output, or is to show you the beginning

The values you set in the dialog box are remembered throughout the editing session, enabling you quickly to re-issue the same command with the minimum of effort.

You can run only one DOS command at a time in this way.

In order to run a DOS command, PFE requires you to have a helper module in the same directory as the main executable. The helper module is responsible for actually running the DOS command and redirecting its output.

For the 32-bit edition the helper module is **$pfeds32.exe**. For the 16-bit edition the module is called **$pfedos.exe**, and has an associated PIF file **$pfedos.pif**. As supplied, this PIF file runs the command in full-screen and exclusive modes; if you are working in 386 enhanced mode, you may wish to use the PIF editor to modify this.

The **Execute Repeat DOS Command To Window** command allows you to repeat the last command you issued with this dialog without needing to specify the details again. Note that if the command line you gave contained substitution points, these will be re-evaluated.

This functionality is not available if you are running the 16-bit Edition in the Win16 subsystem of Windows NT.

Substituting The Current File Name In Commands

Substituting The Current File Name In Commands

PFE gives you a quick way to include the name of the file showing in the current window in the command lines specified in the dialogs you start with the **Execute DOS Command To Window** and **Execute Launch Application** commands.

You mark a place in the command line where part of the full path name is to be substituted with a **"%"** character, followed by a letter to show which part you require. You can specify any of these substitutions:

%d	The directory part of the path name, without a trailing "\"
%e	The extension part of the path name, without a leading "**.**"
%f	The filename and extension parts of the path name, separated by a "**.**"
%n	The filename part of the path name
%p	The entire path name
%u	The drive part of the path name, followed by a "**:**"
%%	A single "**%**" character

For example, suppose that the file showing in the current window was named `c:\software\graphit\source\main.c`

The command line

`nmake %n.obj`

would be expanded by PFE to be

`nmake main.obj`

and the command line

`dir %u%d\*.c`

would expand into

`dir c:\software\graphit\source\*.c`

If you don't have a file open, or if the current window does not have a file name associated with it - for instance, if it is a window showing command output, or was created with a **File New** command and you have not yet associated a file name with it - substitution will fail and you will see an error message.

The expanded command line must also not be longer than 512 bytes in total size. Note that this allows you to generate command lines that are longer than will be accepted by either the operating system or the command processor program that is asked to run them.

Launching Windows Applications

The **Execute Launch Application** command gives you a convenient way of starting another Windows or DOS application running. If you're developing a program, you'll find it a useful for starting up the latest version quickly.

Applications started in this way run independently of PFE. To run a DOS application such as a compiler, and see its output in an edit window, see the section on Running A Command And Capturing Output.

The command starts a dialog that lets you specify various things about the application you want to run. You can set:

- The command line to be executed. You can have PFE substitute parts of the name of the file showing in the current window automatically for you if you wish.

- The working directory to be used. PFE will make this the current directory for the command process before the command is issued (PFE's own current directory setting is not affected)

- Whether PFE is to minimize itself into an icon once the command has been issued

- Whether PFE is automatically to save any altered files before starting the task

The values you set in the dialog box are remembered throughout the editing session, enabling you quickly to start the same application again with the minimum of effort.

If you check the **Save Changed Files** box, PFE will check whether any of the files that you are editing have been altered, and will ask you if you want to save them. You should take care if you reply **Yes** to the question, as *all* the files that have been altered will be saved. Note that windows holding un-named files are not checked for changes.

You can launch as many independent applications as you wish.

Windows Tools

PFE allows you easily to launch Windows tools, such as the Dialog Editor, the Image Editor, and the CodeView debugger, by name from a list that you can configure.

When you add a Windows tool to the list you assign it a *tool name*, which can be any string up to 32 characters long. You associate with this name the command string you want to execute and the initial directory; then, when you want to start the tool, simply pick the name from the list.

You can have as many tool names referring to an application as you want: you could thus have a set that each invoke an application with slightly different command-line arguments.

Adding A Windows Tool Entry
Changing A Windows Tool Entry
Deleting A Windows Tool Entry

Launching A Windows Tool

Adding A Windows Tool Entry

To add an entry to the list of Windows Tools, you need to use the **Execute Configure Windows Tools** command. This starts a dialog that lets you manipulate the list of tools.

For each tool name, you can specify:

- The command line to be executed

- Whether PFE is to make itself into an icon before launching the tool

- Whether is PFE is automatically to save any changed files before launching the application

The Execute Configure Windows Tools Dialog

Changing A Windows Tool Entry

To change an entry in the list of Windows Tools, you need to use the **Execute Configure Windows Tools** command. This starts a <u>dialog</u> that lets you manipulate the list of tools.

For each tool name, you can specify:

- The command line to be executed

- Whether PFE is to make itself into an icon before launching the tool

- Whether is PFE is automatically to save any changed files before launching the application

Deleting A Windows Tool Entry

To delete an entry from the list of Windows Tools, you need to use the **Execute Configure Windows Tools** command. This starts a dialog that lets you manipulate the list of tools.

Launching A Windows Tool

To launch a tool that you've defined in the list of Windows tools, use the **Execute Launch Windows Tool** command. This starts a <u>dialog</u> that lets you select one of the tools you've configured.

You can launch the tool with the details you set when you configured it with the **Execute Configure Windows Tool** command, or you can change any details you wish.

PFE records the details that you set, and uses these as the default when you launch this tool again; however, you can return to the configured settings with a single button click.

The application that you launch runs independently of PFE.

Controlling PFE Over A DDE Link

Unless you started it with the command line option "**/m**", or used the **Options Preferences** command to specify that PFE should run multiple instances, PFE will act by default as a **DDE Server**, allowing you to control its actions from some other program. This program will need to have the ability to act as a **DDE Client**; many standard applications already support this.

If you are using an existing Windows application as a DDE client you will need to consult its documentation for how to perform the necessary operations. You can easily write your own client application in Visual Basic (the DDETester application supplied with PFE demonstrates many of the techniques you will need); or you can write at API level using the Microsoft **DDEML** support library described in the SDK documentation and third party books.

Depending on how many DDE commands your application needs to send to PFE, and how frequently, it may be adequate for it to operate by establishing a DDE connection, issuing the command, and then closing the connection.

If, though, the application is going to make heavy use of DDE, it will be more efficient for it to establish a conversation once and leave it established until it has finished. If PFE detects that there are DDE conversations in progress when the user tries to exit from it, it will show a dialog warning that a client application is still operating, and asking whether the user wishes to abandon the closedown.

Starting And Stopping A DDE Server
Issuing Commands Over A DDE Link
Requesting Data Over A DDE Link
Pasting Data Over A DDE Link
Opening Files Over a DDE Link

In this release of PFE, the DDE interfaces are still under development, and there may be considerable changes from what is described here in future versions.

Starting And Stopping A DDE Server

Normally, when you start PFE, it will run as a DDE server automatically. However, if you start PFE running in multiple-instance mode, either with the **"/m"** command line option, or by permitting multiple instances with the **Options Preferences** command, it will not automatically act as a server.

If you do want an instance of PFE that you started in this way to act as a DDE server, you can use the **Options DDE Server** command to activate server mode. You can also use this command to turn the server off; the menu item will have a check mark beside it when the server is running.

Normally, you should try to ensure that only one instance of PFE is running as a DDE server at any time. Although it is permissible to run multiple servers, you can't then predict which server an application will connect to at any time. There is thus the real chance that an application may send DDE commands to the wrong PFE instance.

Issuing Commands Over A DDE Link

DDE commands are sent to PFE using the standard DDE methods. Your application should open a DDE link to PFE specifying the relevant DDE service name, and a topic name of either "**Editor**" or "**System**". From Visual Basic for example, you would perform a *Link.Execute* operation. The DDE item name is not used.

Once the link is established, you can send one or more command strings to be executed. The command string should have the general form

```
[CommandName(argument,...)]
```

where the arguments you supply within the parentheses will vary depending on the command. The command name can be written in any mixture of upper- and lower-case letters. A maximum of 128 bytes can be sent in any transaction.

PFE will return a response of **DDE_FACK** immediately if the command syntax is ostensibly correct, or **DDE_FNOTPROCESSED** if there is some obvious error. The command itself is executed asynchronously to the DDE client, which will regain control immediately after sending it. If the last DDE command resulted in a lengthy action - such as loading a large file, or asking the user for input in a dialog box - the server may not be able to accept a new command immediately. In this case it will return a reply **DDE_FBUSY** to the client, which should wait for a suitable interval before attempting the command again.

To determine whether any DDE link command worked or not you need to request the data for the item "**Result**" from the "**Editor**" topic. This data item will contain the string "**OK**" if the command succeeded, "**Error**" if it failed, and "**Busy**" if the server has not yet completed its processing.

List Of DDE Commands

Requesting Data Over A DDE Link

The PFE DDE server allows you to request items of data from your application. For example, you can ask PFE whether the DDE server is busy, or what the result of the last DDE command was; you can also find out about the state of files that are being edited and so on.

To request data, you need to open a DDE link to the relevant <u>DDE service name</u> with an appropriate topic. The link item is the name of the information that you require. From Visual Basic, for example, you would perform a *Link.Request* operation. The reply will always be a string of characters: numbers are converted into ASCII digits.

<u>List Of DDE Data Items</u>

Pasting Data Over A DDE Link

PFE gives you several ways of pasting data into the current file over a DDE link.

Using the EditInsert() command

For small amounts of data, you can send the <u>DDE command</u> **EditInsert()**, giving the string to be pasted as the argument. This will often be the most convenient technique to use.

Using the clipboard

With this method, you should first place the data to be pasted into the Windows clipboard in **CF_TEXT** format, with the end of each line marked by a CR-LF bytes pair and a null byte marking the end of data.

Then, you should issue the DDE command **EditPaste()**.

The command will run asynchronously to the client application. To determine whether it succeeded or failed, you will need to <u>request the data</u> item "**Result**" from the "**Editor**" topic.

Using DDE Poke

With this method, you initiate a conversation with the PFE DDE server using the relevant <u>DDE service name</u>, the topic "**Editor**" and item "**Paste**".

Your application should first place the data to be pasted into a shareable global memory segment in **CF_TEXT** clipboard format, where each line is ended by a CR-LF byte pair, and a final null byte marks the end of the data.

It should then perform a DDE **Poke** operation passing the handle to the memory segment. From Visual Basic, for example, you would execute a *Link.Poke* command.

PFE will reply **DDE_FACK** if the command was accepted, **DDE_FNOTPROCESSED** if there was an error, and **DDE_FBUSY** if the server was busy and could not accept the command.

With the 16-bit edition you may not paste more than 64 kilobytes of data in a single operation. With the 32-bit edition there is no effective limit.

The paste operation will normally run asynchronously to the client application. To determine whether it succeeded or failed, you will need to <u>request the data</u> item "**Result**" from the "**Editor**" topic.

Opening Files Over A DDE Link

PFE gives you two distinct ways of opening files across a DDE link.

Using DDE Commands

The DDE <u>commands</u> **FileOpen()** and **FileView()** each allow you to open a single file, for editing or in read-only mode respectively. For opening small numbers of files, it will be most convenient to use this method.

The file names you give in these commands may include the normal DOS wildcard characters.

Using DDE Poke

To open a larger number of files, you can perform a DDE Poke operation, passing all the names in a single block of memory. To do this, you should

- Create a suitably large block of shareable global memory (under Windows 3.1, this block must not exceed 64 kilobytes in size)

- Place the name of the directory to be used by default with each file name at the start of the block, followed by a single space.

- Place the names of all the files following this, separating the names from each other by a single space. Files that are in the default directory should be stored without a directory path.

- Terminate the block with a single null byte.

- Perform a DDE Poke operation using the topic "**Editor**" and item name "**OpenFiles**" to open the files for editing, or the item name "**ViewFiles**" to open them in *read-only* mode.

PFE will open each file in turn for editing. If a file name does not start with a drive specification or a "\" character, the name of the default directory specified at the start of the block will be prepended to it; otherwise, the file name will be used exactly as presented.

File names may contain the normal DOS wildcard characters.

For example, a data block that contained the string

```
 c:\source main.c c:\windows\win.ini ..\std.h *.h
```

would open the files **c:\source\main.c**, **c:\windows\win.ini**, **c:\source\..\std.h** and all the files matching the wildcard pattern **c:\source*.h**

Reference Section

This section of the help file contains reference material on PFE.

Command Line Options
Dictionary Of Functions
Default Key Mappings
DDE Commands And Data Items
Language Awareness

Command Line Options

When you start PFE, you can give several options on the command line to control its actions. These options should follow the name of the PFE executable itself, and precede any file names.

RUN MODE OPTIONS

These options control PFE's *run mode*, or whether more than one instance of it will run at a time. You can set the default run mode with the **Options Preferences** command, but you can override that value when you start PFE with a command line option.

You can give one of the following options to control the run mode, overriding the initialisation file:

/m   If the option is given, the command will start a new instance of PFE whether any previous instance is running or not. If it is absent, PFE will attempt to pass control to any other instance that is running as a DDE server, and will start as a separate instance only if this fails.

            An instance of PFE started with this option present will not act as a DDE server unless the "**/d**" option was also used.

/s   If this option is given, the command will look for any previously-running instance, and will activate it instead of running a new one. If you've given any files on the command line, at least one running instance must also be acting as a DDE server.

            If no instance is already running, or if it can't be activated, a new instance will start.

CONFIGURATION OPTIONS

These options affect how PFE configures itself when it starts up:

/d   If this option is given, an instance of PFE started with the "**/m**" option (or with an option that implies the "**/m**" option) will act by default as a DDE server; without it the instance will not have the server enabled when it starts.

/i directorypath

            This option specifies a directory path name that identifies the directory in which PFE should look for its initialisation file and default key map file. This switch overrides the default value (the Windows directory) or any location set with the **Options Preferences** command.

            The directory path must not have a trailing "\" character unless it specifies the root directory of a drive

            Using this option implies the "**/m**" option; a new instance of PFE will be started whether or not any others are running, and will not by default run as a DDE server.

/k pathname

            This option specifies a key map file to be loaded on start-up instead of the **pfe.key** or **pfe32.key** file in the initialisation directory. **Pathname** can specify any valid key map file; if it is the single character "**-**" (minus) PFE will not load any key mapping at all, but will use the built-in default values described in this help file.

            Using this option implies the "**/m**" option; a new instance of PFE will be started whether or not any others are running, and will not by default run as a DDE server.

Whatever value you give for **pathname**, using this option will disable *all* use of the default **pfe.key** file. The **Save** button in the <u>key mapping dialog</u> will not be available when you change the mapping. To save to this file,   you will need to use the **Save As** button and give the name explicitly.

OTHER OPTIONS

/g position

If this option is given, PFE will automatically move the caret to the specified line number and character location in the *first* file named in the command line. If the first file named contains a wildcard character, the option is ignored

The position value can be a simple line number, or any non-relative <u>file position specification</u>.

/v

If this option is given, PFE will open any files that you name on the command line in *read-only* mode.

/w

If this option is given, PFE will check the Most Recently Used files list to see if it can open files on the command with windows at the position they had the last   time they were used, regardless of whether use of positions for command line files has been disabled by the **<u>Options Preferences</u>** command

Examples

A command line like this

```
pfe data.txt results.txt
```

will start PFE, or activate a previously-running instance that was acting as a DDE server, and open the files **data.txt** and **results.txt** in the current directory.

The line

```
pfe /v data.txt results.txt
```

would act in the same way; but the files would be opened in read-only mode.

The command line

```
pfe /g27/5 cat.c dog.c
```

will open files **cat.c** and **dog.c**. The caret in the window showing **cat.c** will be automatically positioned at window column 5 of line 27; the caret in the window shoing **dog.c** will be positioned at the start of the file.

The command line

```
pfe /m
```

will start a new instance of PFE, whether or not any others were also running. The DDE server will not be enabled when the instance starts.

The command line

```
pfe /k mapping.key
```

will start a new instance of PFE and load the key mappings contained in the file **mapping.key** in the current directory. The DDE server will not be enabled when the instance starts.

The command line

```
pfe /i c:\project
```

will start a new instance of PFE and look in directory **c:\project** for the initialisation file and default key map file. The DDE server will not be enabled when the instance starts.

Dictionary Of Functions

PFE allows you to map keyboard keys or key sequences to the following functions, which appear in the functions list of the key mapping dialog.

Not all the functions that you can map to keys have equivalent menu commands; those that do have **[*]** shown following the description.

CaretBottomOfWindow   Moves the caret to the start of the lowest line visible in the window

CaretBottomOfWindowSelect
                                 Moves the caret to the start of the lowest line visible in the window, extending the selection

CaretDown   Moves the caret one line down

CaretDownSelect   Moves the caret one line down, extending the selection

CaretEndOfLine   Moves the caret to the end of the current line

CaretEndOfLineSelect   Moves the caret to the end of the current line, extending the selection

CaretEndOfFile   Moves the caret to the end of the current file

CaretEndOfFileSelect   Moves the caret to the end of the current file, extending the selection

CaretLeft   Moves the caret one character to the left

CaretLeftSelect   Moves the caret one character to the left, extending the selection

CaretLeftWord   Moves the caret one word to the left

CaretLeftWordSelect   Moves the caret one word to the left, extending the selection

CaretPageDown   Moves one page down in the current file

CaretPageUp   Moves one page up in the current file

CaretRight   Moves the caret one character to the right

CaretRightSelect   Moves the caret one character to the right, extending the selection

CaretRightWord   Moves the caret one word to the right

CaretRightWordSelect   Moves the caret one word to the right, extending the selection

CaretStartOfFile   Moves the caret to the start of the current file

CaretStartOfFileSelect   Moves the caret to the start of the current file, extending the selection

CaretStartOfLine   Moves the caret to the start of the current line

CaretStartOfLineSelect   Moves the caret to the start of the current line, extending the selection

CaretStartOfText   Moves the caret to the first non-white-space character in the current line

CaretStartOfTextSelect	Moves the caret to the first non-white-space character in the current line, extending the selection
CaretTopOfWindow	Moves the caret to the start of the top line in the window
CaretTopOfWindowSelect	Moves the caret to the start of the top line in the window, extending the selection
CaretUp	Moves the caret one line up
CaretUpSelect	Moves the caret one line up, extending the selection
EditCancelSelection	Removes any highlight from selected text in the current window
EditClearUndo	Forgets the details of all edit actions performed up to this point on the current file, freeing off the memory used to hold them **[*]**
EditCopy	Copies the selected text to the clipboard **[*]**
EditCut	Copies the selected text to the clipboard and deletes it **[*]**
EditDeleteSelection	Deletes the selected text
EditDeleteBackwards	Deletes the character to the left of the caret
EditDeleteForwards	Deletes the character to the right of the caret
EditDeleteLine	Deletes the whole of the line that the caret is in **[*]**
EditDeleteToEndOfLine	Deletes from the position of the caret to the end of the current line **[*]**
EditGotoLine	Starts a dialog that prompts for the line number to go to in the current file **[*]**
EditInsertHardTab	Inserts a TAB character, whether or not the current window's Window Mode is set for soft tabbing
EditInsertSoftTab	Inserts the required number of spaces to bring the caret to the next TAB stop position, even if the current window's Window Mode is not set for soft tabbing
EditInsertTab	Inserts either a TAB character or the required number of spaces to bring the caret to the next TAB stop position, depending on whether the current window's Window Mode is set for hard or soft tabbing
EditMarkUnchanged	Removes PFE's indication that the current file has been altered
EditNewLine	Inserts a new line character at the position of the caret
EditPaste	Pastes the contents of the clipboard into the current file at the position of the caret **[*]**
EditSearch	Searches for a string in the current file **[*]**
EditSelectLine	Selects all the characters in the line that the caret is within, moving the

caret to the start of line and scrolling the window, if necessary, to make it visible

EditSelectWord   Selects the whole word that the caret is currently within **[*]**

EditShowCaret   Adjusts the position of the text in the current window so that the caret is as close to the central line as possible **[*]**

EditShowNextLine   Scrolls the window upwards to show the next line in the file. The caret remains in the same place within the file

EditShowNextPage   Scrolls the window upwards to show the next page in the file. The caret remains in the same place within the file

EditShowPreviousLine   Scrolls the window downwards to show the previous line in the file. The caret remains in the same place within the file

EditShowPreviousPage   Scrolls the window downwards to show the previous page in the file. The caret remains in the same place within the file

EditSplitLine   Splits the current line at the position of the caret, leaving the caret unmoved

EditRepeatLastFind   Repeats the last find operation **[*]**

EditRepeatLastReplace   Repeats the last replace operation **[*]**

EditReplace   Replaces occurrences of one string with another string **[*]**

EditTextIndent   Indents the current line, or all the lines in a selection, by one tab stop **[*]**

EditTextInsertASCIICode   Starts a dialog to insert a character specified by its ASCII code **[*]**

EditTextLowercaseSelection

   Converts all the letters in the current selection to lower case **[*]**

EditTextMatchBrace   Moves the caret to the brace character matching the one currently under it, having regard to language syntax **[*]**

EditTextMatchBraceSelect   Moves the caret to the brace character matching the one currently under it, having regard to language syntax, and highlights all the text between and including the braces **[*]**

EditTextTransposeCharacters

   Exchanges the character under the caret with the one to its left **[*]**

EditTextUndent   Undents the current line, or all the lines in a selection, by one tab stop **[*]**

EditTextUppercaseSelection

   Changes all the text that is currently selected to be upper case **[*]**

EditTextWidenBraceSelect   Highlights all the text between and including the first matching pair of brace characters that includes either the currently-highlighted text, or the character under the caret if no text is highlighted **[*]**

EditUndo	Undoes the last edit action you performed **[*]**
ExecConfigureTools	Starts a dialog to configure details of program development tools **[*]**
ExecControlPanel	Starts the Windows Control Panel **[*]**
ExecDOSCommand	Runs a DOS command, capturing output in an edit window **[*]**
ExecDOSPrompt	Runs a DOS command-line shell **[*]**
ExecLaunchApp	Starts a Windows or DOS application to run independently of PFE **[*]**
ExecLaunchTool	Starts a program development tool **[*]**
ExecFileManager	Runs the Windows File Manager **[*]**
ExecPrintManager	Runs the Windows Print Manager **[*]**
ExecRepeatDOSCommand	Repeats the last DOS command run with the **ExecDOSCommand** function **[*]**
FileAbandon	Closes the file being edited in the current window, discarding any unsaved changes
FileClose	Closes the file being edited in the current window, prompting if there are unsaved changes **[*]**
FileCloseAll	Closes all the files being edited, prompting if any of them contain unsaved changes **[*]**
FileExit	Terminates the PFE editing session **[*]**
FileInsert	Inserts a file into the current one at the position of the caret **[*]**
FileMail	Mails the current file **[*]**
FileName	Changes the name of the file associated with the current window **[*]**
FileNew	Creates a new empty edit window for an unnamed file **[*]**
FileOpen	Opens an existing file for editing **[*]**
FilePrint	Prints the current file **[*]**
FilePrintSetup	Runs a dialog that allows printer information to be configured **[*]**
FileSave	Saves the current file to disk, provided that it has been altered **[*]**
FileSaveAs	Saves the current file to disk, allowing the output file name to be specified **[*]**
FileSaveAll	Saves all altered files and templates **[*]**
FileView	Opens an existing file in *read-only* mode **[*]**

FileWrite	Writes a file to disk whether or not it has been changed, allowing you to specify the name of the output file **[*]**
HelpAbout	Shows PFE's *About* box **[*]**
HelpCommands	Displays the PFE Help File at the **Commands** topic **[*]**
HelpContents	Displays the PFE Help File at the **Contents** topic **[*]**
HelpContextHelp	Starts the Windows help engine to give help on either the highlighted text in the current window, or the word under the caret **[*]**
HelpOnHelp	Starts the Windows help engine to give help about itself **[*]**
HelpProcedures	Displays the PFE Help File at the **Procedures** topic **[*]**
HelpReference	Displays the PFE Help File at the **Reference** topic **[*]**
HelpUserDefined1	Starts the Windows help engine, using the first help file configured with the **Options Preferences** command **[*]**
HelpUserDefined2	Starts the Windows help engine, using the second help file configured with the **Options Preferences** command **[*]**
HelpUserDefined3	Starts the Windows help engine, using the third help file configured with the **Options Preferences** command **[*]**
HelpUserDefined4	Starts the Windows help engine, using the fourth help file configured with the **Options Preferences** command **[*]**
HelpUserDefined5	Starts the Windows help engine, using the fifth help file configured with the **Options Preferences** command **[*]**
MacroReplay	Replays the last recorded keyboard macro **[*]**
MacroStartRecorder	Starts recording keystrokes and menu selections **[*]**
MacroStopRecorder	Stops recording keystrokes and menu selections **[*]**
OptionsCurrentModes	Starts a dialog to change the current window and file modes **[*]**
OptionsDDEStart	Starts the DDE server
OptionsDDEServer	Stops the DDE server
OptionsDDEToggle	Starts the DDE server if it is not running, and stops it if it is **[*]**
OptionsDefaultModes	Starts a dialog to change the window modes for specific file types **[*]**
OptionsKeyMappings	Starts a dialog to change the mapping of keys to functions **[*]**
OptionsPreferences	Starts a dialog to configure how you wish PFE to operate **[*]**
OptionsPrefixKeys	Starts a dialog that allows the user to define what prefix keys are active

OptionsResetModes	Sets the window modes for the current window, and the file modes for the file that it is showing, to those appropriate for the file's type **[*]**
OptionsScreenFontAnsi	Sets the screen font to be the standard non-proportional ANSI font **[*]**
OptionsScreenFontOEM	Sets the screen font to be the standard OEM font **[*]**
OptionsScreenFontOther	Starts a dialog that allows the user to select any screen font **[*]**
OptionsScreenFontSystem	Sets the screen font to be the standard non-proportional system font **[*]**
OptionsToggleInsertMode	Switches between *insert* and *overwrite* modes
OptionsToggleLineNumbers	Turns line numbering on or off in the current window
OptionsToggleStatusBar	Makes the status bar visible or invisible **[*]**
OptionsToolbarBottom	Positions the tool bar at the bottom of the screen **[*]**
OptionsToolbarFloat	Makes the tool bar into a detached floating window **[*]**
OptionsToolbarHide	Makes the tool bar invisible **[*]**
OptionsToolbarLeft	Positions the tool bar at the left of the screen **[*]**
OptionsToolbarRight	Positions the tool bar at the right of the screen **[*]**
OptionsToolbarShow	Makes an invisible tool bar visible at its previous location **[*]**
OptionsToolbarTop	Positions the tool bar at the top of the screen **[*]**
SysSetMenuMode	Highlights the leftmost menu item on the main window's menu bar, so you can easily access menu items from the keyboard. The effect is the same as pressing and releasing the **Alt** key.
TemplateFileAttach	Attaches a template file so the templates within it can be used **[*]**
TemplateFileCreate	Creates a new, empty template file **[*]**
TemplateFileDetach	Detaches an attached template file **[*]**
TemplateFileSave	Saves an altered template file to disk **[*]**
TemplateDelete	Deletes a template from a template file **[*]**
TemplateEdit	Edits a template from within a template file **[*]**
TemplateFindMark	Searches in a forward direction from the position of the caret for the next occurrence of a *template marker* **[*]**
TemplateInsert	Inserts a template from a template file into the current file at the position of the caret **[*]**
TemplateInsertMark	Inserts a *template marker* into a template **[*]**

TemplateNew	Creates a new empty window suitable for editing a template **[*]**
TemplateStore	Stores a template in a template file **[*]**
TemplateStoreAs	Stores a template in a template file, allowing its name to be changed **[*]**
WindowArrangeIcons	Arranges all iconized windows neatly **[*]**
WindowCascade	Arranges all non-iconic windows in a cascade **[*]**
WindowClose	Closes the current window **[*]**
WindowDuplicate	Makes an exact duplicate of the current window **[*]**
WindowIconizeAll	Makes all existing windows iconic **[*]**
WindowMaximize	Maximizes the current window **[*]**
WindowNext	Switches control to the next non-iconic window **[*]**
WindowRestore	Restores the current window from iconized or maximized state
WindowTileHorizontal	Arranges all non-iconic windows in a tile pattern, maximizing their depth **[*]**
WindowTileVertical	Arranges all non-iconic windows in a tile pattern, maximizing their width **[*]**
WindowWiden	Adjusts the size and position of the current window to make it as wide as possible **[*]**

The Default Key Mappings

As supplied, PFE contains a set of key mappings that enable you to perform the most common operations quickly. You may use these as they stand, or may <u>modify</u> them to suit your preferences.

By default, no prefix keys are enabled, so that only single keys are available for mapping.

<u>Description Of Default Key Mappings</u>
<u>Table Of Keys And Default Functions</u>

Description Of Default Key Mappings

This section lists the keys that have default mappings, giving you a description of the action that they carry out and the equivalent menu command (if any).

For a table that lists the keys with the names of the functions used in the key mapping dialog, see the **Table Of Keys And Default Functions**.

FUNCTION KEYS

F1	Opens the PFE Help File at the **Contents** topic
Shift+F1	Executes the **Help Screen/Menu Help** command to enter interactive help mode
Ctrl+F1	Executes the **Help Context Help** command to give help on the highlighted text in the current window or the word under the caret
F2	Executes the **Edit Find** command to search for a string
Shift+F2	Executes the **Edit Repeat Last Find** command to repeat the last search operation
F3	Executes the **Edit Replace** command to replace strings
Shift+F3	Executes the **Edit Repeat Last Replace** command to repeat the last replace operation
F4	Executes the **Template Find Mark** command to search for a template mark
Shift+F4	Executes the **Window Tile Horizontal** command to arrange windows in a tile pattern
Alt+F4	Executes the **File Exit** command to end your PFE session
Shift+F5	Executes the **Window Cascade** command to arrange windows in a cascade pattern
Ctrl+F5	Executes the **Edit Show Caret** command to bring the line containing the caret to the centre of the window
F6	Executes the **Template Insert Mark** command to insert a template mark
F7	Executes the **Macro Replay** command to replay a keyboard macro
Shift+F7	Executes the **Macro Start Recorder** command to begin recording a keyboard macro
Ctrl+F7	Executes the **Macro Stop Recorder** to stop recording a keyboard macro
F9	Executes the **Template Insert** command to insert a template into the current window
F11	Executes the **Execute DOS Command To Window** command to run a DOS command and capture its output in a window
Shift+F11	Executes the **Execute Launch Application** command to launch a Windows application to run independently of PFE

Ctrl+F11	Executes the **<u>Execute DOS Prompt</u>** command to start a DOS session
F12	Executes the **<u>Execute Launch Windows Tool</u>** command to launch a tool from the list of configured Windows Tools
Shift+F12	Executes the **<u>Execute Configure Windows Tools</u>** command to let you configure the list of Windows Tools

MOVEMENT KEYS

For these keys, the default mapping is such that adding **Shift** to the combination causes a selection to be extended.

Note that where the **Alt** key is combined with either an arrow key, or one of **Home**, **End**, **PgDn**, **PgUp**, **Ins** or **Del**, you must use the keys in the extended key areas, and not those in the numeric keypad.

Up	Moves the caret up by one line
Shift+Up	Moves the caret up by one line, extending the selection
Down	Moves the caret down by one line
Shift+Down	Moves the caret down by one line, extending the selection
Left	Moves the caret left by one character
Shift+Left	Moves the caret left by one character, extending the selection
Ctrl+Left	Moves the caret left by one word
Ctrl+Shift+Left	Moves the caret left by one word, extending the selection
Right	Moves the caret right by one character
Shift+Right	Moves the caret right by one character, extending the selection
Ctrl+Right	Moves the caret right by one word
Ctrl+Shift+Right	Moves the caret right by one word, extending the selection
PgDn	Moves the caret down by one page
Shift+PgDn	Moves the caret down by one page, extending the selection
Ctrl+PgDn	Moves the caret to the start of the last line in the window
Ctrl+Shift+PgDn	Moves the caret to the start of the last line in the window, extending the selection
PgUp	Moves the caret up by one page
Shift+PgUp	Moves the caret up by one page, extending the selection
Ctrl+PgUp	Moves the caret to the start of the first line of the window

Ctrl+Shift+PgUp	Moves the caret to the start of the first line of the window, extending the selection
Home	Moves the caret to the start of the line
Shift+Home	Moves the caret to the start of the line, extending the selection
Ctrl+Home	Moves the caret to the start of the file
Ctrl+Shift+Home	Moves the caret to the start of the file, extending the selection
Alt+Home	Moves the caret to the first non-white-space character in the current line
Alt+Shift+Home	Moves the caret to the first non-white-space character in the current line, extending the selection
End	Moves the caret to the end of the line
Shift+End	Moves the caret to the end of the line, extending the selection
Ctrl+End	Moves the caret to the end of the file
Ctrl+Shift+End	Moves the caret to the end of the file, extending the selection

CONTROL KEYS

Ctrl+B	Executes the **Edit Text Match Brace** command to move the caret to the brace character matching the one currently under it
Ctrl+Shift+B	Executes the **Edit Text Match Brace Select** command to move the caret to the brace character matching the one currently under it and highlight all the text between and including the brace characters
Ctrl+C	Executes the **Edit Copy** command to copy selected text to the clipboard
Ctrl+G	Executes the **Edit Goto Line** command to move the caret to a specific line
Ctrl+H	Deletes the character to the left of the caret
Ctrl+I	Inserts a TAB character, or the required number of spaces to bring the caret to the next TAB stop, depending on whether the current window's Window Mode is set for hard or soft TABs.
Ctrl+K	Executes the **Edit Delete To End Of Line** command to delete everything from the caret position to the end of the line
Ctrl+Shift+K	Executes the **Edit Delete Line** command to delete the entire line that the caret is in
Ctrl+N	Executes the **File New** command to create a new, empty edit window
Ctrl+O	Executes the **File Open** command to open an existing file
Ctrl+Shift+O	Splits the current line at the position of the caret, leaving the caret unmoved

Ctrl+P	Executes the **File Print** command to print the file showing in the current window
Ctrl+Q	Executes the **Edit Text Insert ASCII Code** command to insert a character specified by its ASCII code number
Ctrl+S	Executes the **File Save** command to save the current file to disk
Ctrl+T	Executes the **Edit Text Transpose Characters** command to transpose the character under the caret with the one to its left
Ctrl+V	Executes the **Edit Paste** command to paste data from the clipboard into the current window
Ctrl+W	Executes the **Window Select** command to let you choose between many open windows
Ctrl+Shift+W	Executes the **Edit Text Widen Brace Select** command to highlight all the text between and including the closest pair of brace characters that encompasses either the currently-highlighted text, or the character under the caret if no text is selected
Ctrl+X	Executes the **Edit Cut** command to cut selected text to the clipboard
Ctrl+Z	Executes the **Edit Undo** command to undo the last edit action

OTHER KEYS

Enter	Inserts a newline
BackSpace	Deletes the character to the left of the caret
Del	Deletes the character to the right of the caret
Shift+Del	Executes the **Edit Cut** command to cut selected text to the clipboard
Ins	Toggles the current window between Insert and Overwrite modes
Shift+Ins	Executes the **Edit Paste** command to paste data from the clipboard into the current window
Ctrl+Ins	Executes the **Edit Copy** command to copy selected text to the clipboard
Keypad 5	Cancels the current selection, removing the highlight
Tab	Inserts a TAB character, or the required number of spaces to bring the caret to the next TAB stop, depending on whether the current window's Window Mode is set for hard or soft TABs.

Table of Keys And Default Functions

These tables show you the functions that are by default mapped to certain keys. For a description of the action of each function, consult the <u>functions dictionary</u>

A descriptive list of the action of each key is in the **<u>Description Of Default Key Mappings</u>** section.

FUNCTION KEYS

KEY	Used Alone	With Shift	With Ctrl	With Alt
F1	HelpContents	HelpScreenMenuHelp	HelpContextHelp	
F2	EditFind	EditRepeatLastFind		
F3	EditReplace	EditRepeatLastReplace		
F4	TemplateFindMark	WindowTileHorizontal		FileExit
F5		WindowCascade	EditShowCaret	
F6	TemplateInsertMark			
F7	MacroReplay	MacroStartRecorder	MacroStopRecorder	
F8				
F9	TemplateInsert			
F10	SysSetMenuMode			
F11	ExecDOSCommand	ExecLaunchApp	ExecDOSPrompt	
F12	ExecLaunchWindows-Tool	ExecConfigureWindows-Tools		

MOVEMENT KEYS

For these keys, the default mapping is such that adding **Shift** to the combination causes a selection to be extended.

Note that where keys are mapped in combinations including the **Alt** key, you must use the keys from the extended keyboard area, and not those in the numeric keypad

KEY	Used Alone	With Shift	With Ctrl	With Ctrl+Shift
Up	CaretUp	CaretUpSelect		
Down	CaretDown	CaretDownSelect		
Left	CaretLeft	CaretLeftSelect	CaretLeftWord	CaretLeftWordSelect
Right	CaretRight	CaretRightSelect	CaretRightWord	CaretRightWordSelect
PgUp	CaretPageUp	CaretPageUpSelect	CaretTopOfWindow	CaretTopOfWindowSelect
PgDn	CaretPageDown	CaretPageDownSelect	CaretBottomOfWindow	CaretBottomOfWindowSelect
Home	CaretStartOfLine	CaretStartOfLineSelect	CaretStartOfFile	CaretStartOfFileSelect
End	CaretEndOfLine	CaretEndOfLineSelect	CaretEndOfFile	CaretEndOfFileSelect

KEY	With Alt	With Alt+Shift
Home	CaretStartOfText	CaretStartOfTextSelect

CONTROL KEYS

KEY	Used Alone	With Shift
Ctrl+B	EditTextMatchBrace	EditTextMatchBraceSelect

Ctrl+C	EditCopy	
Ctrl+G	EditGotoLine	
Ctrl+H	EditDeleteBackwards	
Ctrl+I	EditInsertTab	
Ctrl+K	EditDeleteToEndOfLine	EditDeleteLine
Ctrl+N	FileNew	
Ctrl+O	FileOpen	EditSplitLine
Ctrl+P	FilePrint	
Ctrl+Q	EditTextInsertASCIICode	
Ctrl+S	FileSave	
Ctrl+T	EditTextTransposeCharacters	
Ctrl+V	EditPaste	
Ctrl+W	WindowSelect	EditTextWidenBraceSelect
Ctrl+X	EditCut	
Ctrl+Z	EditUndo	

OTHER KEYS

KEY	Used Alone	With Shift	With Ctrl	With Ctrl+Shift
Enter	EditNewline			
BackSpace	EditDeleteBackwards			
Del	EditDeleteForwards	EditCut		
Ins	OptionsToggleInsertMode	EditPaste	EditCopy	
Keypad 5	EditCancelSelection			
Tab	EditInsertTab			

PFE uses an initialisation file to record the values that should be carried over from one session to another, and to hold the customisation values you set with the **Options Preferences** command.

The initialisation file is called **pfe.ini** for the 16-bit Edition and **pfe32.ini** for the 32-bit Edition.

In both environments the file is held in your Windows directory by default; you can customise the location with the **Options Preferences** command.

Note that from the next release of the 32-bit Edition initialisation data will be held in the Windows registry rather than in an initialisation file.

DDE Commands And Data Items

PFE can be controlled remotely by another application using a DDE link. PFE supports the standard DDE techniques, and can be used from any compliant application.

There are two ways of communicating with PFE. The application can send it **commands**, which cause PFE to take some action; or it request **data items**, which cause PFE to return it some information.

Controlling PFE Over A DDE Link

DDE Commands
DDE Data Items
DDE Poke Items

DDE Commands

The DDE commands currently supported are listed below. For details of how to use a DDE link to control PFE, see Controlling PFE Over A DDE Link

CaretBottomOfWindow(*select*)

    Moves the caret to the start of the last line in the window, extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretDown(*count,select*)

    Moves the caret down by *count* lines. If *select* is 1 the characters it moves over are selected and shown in reverse video; if it is 0 they are not and any outstanding selection is cancelled.

CaretEndOfLine(*select*)

    Moves the caret to the end of the current line,   extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretEndOfFile(*select*) Moves the caret to the end of the file, possibly extending a selection.

CaretHome(*select*) Moves the caret to the start of the current line, extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretLeft(*count,select*)

    Moves the caret left by *count* characters,   extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretLeftWord(*count,select*)

    Moves the caret left by *count* words,   extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretRight(*count,select*)

    Moves the caret right by *count* characters,   extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretRightWord(*count,select*)

    Moves the caret right by *count* words,   extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretStartOfFile(*select*)

    Moves the caret to the start of the file, possibly extending a selection.

CaretTopOfWindow(*select*)

    Moves the caret to the start of the first line in the window, extending the selection if *select* is 1, and not extending it if *select* is 0.

CaretUp(*count,select*)

    Moves the caret up by *count* lines, extending the selection if *select* is 1, and not extending it if *select* is 0.

ComeToForeground()

    Makes PFE the active application and brings its window to the foreground. If the main window is iconic, it will first be restored.

EditCopy() Copies the currently selected text to the clipboard.

EditCut()   Deletes the currently selected text, copying it to the clipboard.

EditDeleteBackwards(*count*)

      Deletes the *count* characters to the left of the caret.

EditDeleteForwards(*count*)

      Deletes the *count* characters to the right of the caret.

EditGotoLine(*position,select*)

      Moves the caret to a position in the file specified by *position*. This may be a simple line number, or an absolute or relative <u>file position specification</u>. If *select* is "**1**", the text from the starting position to the start of the target line will be highlighted.

EditFind(*"string",options*)

      Searches the current window for the specified *string*. Within *string*, special characters are represented with the same <u>notation</u> as used in the dialog started by the **Edit Find** command.

      The *options* parameter is a numeric value that specifies how the search operation is to be carried out, made up by summing any values from this list:

        1 Search in the reverse direction
        2 Search is to be case sensitive
        4 Highlight the matching string
        8 Highlight all text from the starting position of the caret to the end of the matching string

      The **Result** data item will return "**OK**" if the search string was found, and "**Error**" if it was not.

EditInsert("*string*")

      Inserts the quoted string into the current file at the position of the caret. Within the string, the characters "**\f**" represent a Form Feed, "**\n**" represent a line break and "**\t**" a tab character. To insert a "****" character, specify it as "****", to insert a quote character, specify it as two consecutive quote characters. The notation "**xnn**", where "**nn**" is two hexadecimal digits, allows you to specify an arbitrary character code (the null value "**\x00**" is not permitted).

EditPaste()   Causes the contents of the Windows clipboard (provided that it is in **CF_TEXT** format) to be pasted into the current file at the position of the caret. You can also use a <u>DDE Poke</u> operation to paste data.

EditSelectLine()   Selects all the characters in the line containing the caret. The caret is moved to the start of the line and the window is scrolled if necessary to make this visible.

FileAbandon()   Closes the current file. If the file has been altered, unsaved changes will be discarded.

FileClose()   Closes the current file. If the file has been altered, the user will see a dialog box asking him if he wishes to save the data.

FileInsert(*"filename"*) Inserts the specified file into the current file at the position of the caret. If

the filename is not an absolute pathname, it is treated as being relative to PFE's current working directory. If the filename is omitted, PFE will show a dialog box to ask the user to supply it.

FileName(*"filename"***)** Changes the name of the file associated with the current window. If the filename is not an absolute pathname, it is treated as being relative to PFE's current working directory. This command will fail if the current window contains a template.

FileNew() Creates a new window for an un-named file.

FileOpen(*"filename",use_saved_position***)**

Opens the specified file for editing. If the filename is not an absolute pathname, it is treated as being relative to PFE's current working directory. You can open several files at once with a DDE Poke operation using the item "**OpenFiles**".

The filename you specify in this command may contain the normal DOS wildcard characters.

If *use_saved_position* is omitted, or is **0**, PFE will open the new window using a default position and size, and will position the caret at the start of the first line. If the value is given as **1**, PFE will examine the list of Most-Recently-Used files to see if the file has been used before. If it has, the window will be opened at the position and size it last had, and the caret will be positioned where it was in the file when it was closed.

FileSave() Writes the current file to disk, providing it has been changed since it was last saved. If the file is currently un-named, the user will see a dialog asking him for the file name it is to be saved under.

FileSaveAs(*"filename"***)** Writes the current file to disk, changing its name in the process. If the filename is not an absolute pathname, it is treated as being relative to PFE's current working directory.

FileView(*"filename"***)** Opens the specified file in read-only mode. If the filename is not an absolute pathname, it is treated as being relative to PFE's current working directory.

The filename you specify in this command may contain the normal DOS wildcard characters.

FileWrite(*"filename"***)** Writes the current file to disk with the specified filename, whether or not it has been changed. The name by which the file is known to PFE, as shown in the window caption, is not altered. If the filename is not an absolute pathname, it is treated as being relative to PFE's current working directory.

WindowActivate(*id***)** Activates the window with the specified *id* value. The *id* is a numeric string, returned by querying the data item **WindowID** when the desired window is current.

DDE Data Items

The supported items that you can request data from over a DDE link are listed below. For details of how to use a DDE link to control PFE, see Controlling PFE Over A DDE Link

Topic "Editor"

ColumnNumber   Gives you the number of the text column containing the caret in the current file, or "**Error**" if there is no file open. The leftmost column is numbered 1.

FileChanged   Tells you whether the current file has been altered since it was last saved. The reply will be the string "**Yes**" if the file has changed; and "**No**" if either the file has not changed, or there is no current file.

FileName   If the current window contains a file that has an associated file name, the reply will be that name. If there is no current window, or the window has no associated file name, it will be the string "**Error**".

FileWritable   Tells you whether the current file can be written or not (*i.e.* that *read-only* mode has not been set). The reply will be "**Yes**" or "**No**", or "**Error**" if no file is open.

LineNumber   Gives you the number of the line in the current file containing the caret, or "**Error**" if there is no file open. Depending on the size of the file, the number may exceed the capacity of a 16-bit integer.

Result   Gives you the result of the last command executed over the DDE link. The reply will be one of the strings "**OK**" or "**Error**" if a result is available, and "**Busy**" if the server is currently processing a DDE link command. The results of commands executed from the keyboard, menu or toolbar cannot be read by this method.

Status   Gives you the current status of the DDE server. The reply will be one of the strings "**Ready**" and "**Busy**".

VersionString   Gives you the version identification of PFE as a string in the form "*x.yy.zzz*" as shown in the dialog started by the **Help About** command.

WindowID   Gives you the unique id number of the current window as a numeric string. This id value can be used to identify the window for the whole of its life; id values are never re-used in any given PFE session. If no window exists, the string "**Error**" is returned.

WindowTitle   Gives you the string that is the caption of the current window

Topic "System"

Formats   A list of the clipboard formats supported by PFE's DDE server. This will be the string "**TEXT**"

SysItems   A list of items supported by PFE under the "**System**" topic, separated by tab characters

Topics   A list of the topics supported by PFE, separated by tab characters

Status   The current status of the DDE server. The reply will be one of the strings "**Ready**" and "**Busy**".

DDE Poke Items

The DDE Poke items currently supported are listed below. For details of how to use a DDE link to control PFE, see Controlling PFE Over A DDE Link

All DDE Poke operations should use the relevant DDE service name and the topic "**Editor**". The data should be passed in a shareable global data segment.

OpenFiles   This topic allows you to open a number of files for editing in one operation. The data should contain the name of the default directory to be used for each file, separated by one space from a list of filenames, also separated by one space. A null byte should mark the end of the list.

If a file name in the list does not contain a directory path, the default directory string will be prepended to it; otherwise it is used exactly as supplied.

The file names given in the list may contain the normal DOS wildcard characters.

ViewFiles   This topic allows you to open a number of files in *read-only* mode in one operation. The details are as for the **OpenFiles** topic above.

Paste   This topic allows you to paste data into the current window at the position of the caret. The data should be in **CF_TEXT** clipboard format, with the end of each line marked by a CR-LF byte pair, and a final null byte terminating the block.

Under Windows 3, you may not paste more than 64 kilobytes of data in one operation.

DDE Poke operations are carried out asynchronously to the client application. To determine whether they succeeded or failed, the client must request the data from the item "**Result**" in the "**Editor**" topic.

Language Awareness

PFE contains some awareness of the format of programming languages, to help you when writing program sources. Currently only the C language is supported.

To select language awareness, you need to specify the language in the <u>window modes</u> for the window you are editing in. You can specify that any file with a given file type should use a particular language automatically, or you can change the current window's language at any time.

When editing files that have language type set to "**C**", PFE provides these facilities:

- A **#** character typed in an otherwise empty line is always moved to column 1

- A closing **}** brace typed in an otherwise empty line is moved to the same column as the matching preceding opening brace **{**, providing that this is the only character in its line. In locating the opening brace, characters within string delimiters, comment delimiters or in pre-processor directives will be ignored.

The **<u>Edit Text Match Brace</u>**, **<u>Edit Text Match Brace Select</u>** and **<u>Edit Text Widen Brace Select</u>** commands are also set to operate using opening braces from the set **{ [** and **(**, and closing braces from the set **}]** and **)**.

The Options Default Modes Dialog

This dialog allows you edit the default <u>modes</u> that PFE applies when you open files and create windows, and to configure the filename patterns and mode values that make up your <u>mode groups</u>

To define a set of default modes, take the following steps:

1 If you want to edit your <u>mode groups</u> or define a new one, click the **Edit Mode Groups** button. This starts a <u>dialog</u> that lets you configure the groups

2 Select what you want to set the default modes for:

- For files whose names match a pattern in one of your mode groups, click the **Files whose extension is in mode group** button and select the mode group from the list below it

- For a file whose names do not match a pattern in any of your mode groups, click the **Files whose extension is not in any mode group** button.

- For windows created to edit <u>templates</u>, click the **Windows showing templates** button

- For windows created to hold the output of DOS commands run by the **<u>Execute DOS Command with Capture</u>** command, click the **Windows showing DOS command output** button

- For windows created with the **<u>File New</u>** command, click the **Windows created with the File New command** button

3 Click the **Edit modes** button to start the <u>dialog</u> that allows you to configure the modes for the selected object

4 Repeat the above steps as often as you wish, then click the **Close** button to exit this dialog

The Edit Mode Groups Dialog

This dialog allows you to create, delete and edit your <u>mode groups</u>, which you use to set default <u>file and window modes</u>

Each mode group consists of a set of up 16 filename patterns, such as ***.c**, **readme.*** or **makefile**. The patterns can contain the normal DOS wildcard characters "**?**" and "*****". Associated with each group of patterns are set of modes, which PFE will apply as the initial settings whenever you open a file whose filename (minus the directory path) matches one of the patterns specified in the group.

You can define up to 32 mode groups in total. When PFE searches the mode groups to see if it can match the name of the file being opened, it looks at the mode groups in the order that they appear in the dialog; within each mode group, it checks the filename against the patterns in the order you have defined them. You can use this ordered search, and the fact that patterns can be wildcarded, to achieve complex discrimination patterns between your files.

Each mode group you define also has a *name*, which you use to specify the group when you are editing the actual modes associated with it. You can use up to 32 characters in the name, and so can make it a meaningful string such as "**C Source Files**" or "**All text files**"

Defining a new mode group

To define a new mode group, take the following steps:

1   In the **New group** edit control at the bottom of the dialog, type the name of the mode group. You can use up to 32 characters.

2   In the **Filename Patterns** edit control, type the filename patterns that the mode group will contain. You can type up to 16 patterns, separated by spaces, commas or semicolons.

Patterns may contain the normal DOS wildcard characters "**?**" and "*****", which will match exactly one, and any number of characters respectively

Note that the directory path part of a filename is not used in a comparison

3   You can then add the new mode group to the existing list. The **Add at top** button adds the definition to the head of the list; the **Add at end** button adds it to the end; and the **Replace** button will replace the details of any mode group selected in the list box at the top of the panel with the new details

Editing an existing mode group

To change the details of an existing mode group, take the following steps:

1   Double-click on the name of the mode group to be edited in the list box at the top of the panel, or select it and click the **Edit** button. The details will appear in the **New group** and **Filename Patterns** edit controls at the bottom of the panel.

2   Change the mode group name and/or the list of extensions as you require

3   Click the **Replace** button to replace the current details of the mode group with the new values. If you prefer, you can also click on the **Add at top** or **Add at end** buttons to make new entries at the head or tail of the list respectively.

Deleting an existing mode group

To delete an existing mode group, take the following steps:

1 Select the mode group name from the list box at the top of the panel

2 Click the **Delete** button to delete it

Re-Ordering mode groups

When you open a file, PFE will search for the extension in the mode groups in the order that they appear in this dialog panel. If you want to change the order of the mode groups you have defined, take the following steps:

1 Select the name of the mode group whose position you want to change in the list box at the top of the panel

2 Click the **Up** button to move it one place up in the list, or the **Down** button to move it one place down

Any changes you make to your mode groups will not be stored until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. Until that time, you can abandon any changes that you don't want to keep by clicking on the **Cancel** button.

The Options Current Modes Dialog

This dialog lets you set the <u>modes</u> that apply to the current window, and the file that's showing in it.

The dialog box contains two major areas. To the left, there is a list box that allows you to select the category of modes you want to configure, and buttons that apply to the process of changing all the options. To the right is an area whose appearance will change depending on the category of modes you're working with.

Steps

1 Select the category of modes you want to configure from the list box on the left. The appearance of the panel on the right will change as you select the category, to show you the options that are relevant to it

2 In the panel on the right, set the various options as you want them. The individual panels are described in detail in the following help screens

3 If you want to apply the values you have changed and save them to the <u>initialisation file</u>, but carry on making further changes, click the **Apply Now** button.

4 If you want to apply the values you have changed, save them to the <u>initialisation file</u>, and close the dialog, click the **OK** button.

5 To close the dialog and abandon any changes you have made, click the **Cancel** button

Depending on what the current window contains, some of the available panels will not be appropriate and will not appear. The full set of panels are:

<u>Display</u>	What is displayed, such as line numbers
<u>File Properties</u>	Properties relating to the file being shown in the window, rather than the window itself
<u>Input</u>	How PFE handles input to the window
<u>Language Awareness</u>	What language awareness should be used for the window
<u>Saving</u>	What actions PFE should take saving the file in the current window
<u>Print Defaults</u>	What default values should be assumed when printing the current window
<u>Text Colours</u>	What colours should be used in the current window
<u>Text Formatting</u>	Text layout options such as tab size

The Options Current Modes *Display* Panel

This panel of the dialog shown by the **Options Current Modes** command controls what is displayed in the current window.

- Check the **Show line numbers** box to have PFE display the line number of each line in the window on the left side of the window

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Options Current Modes *File Properties* Panel

This panel of the dialog shown by the **Options Current Modes** command controls some properties relating to the file displayed in the current window.

The **Editing options** area contains options related to how you edit files:

- Check the **File is read only in all windows** box to have the file set into *read only* mode. You will not be able to change the file in any window showing it until you change the setting again, either with this command, or by double-clicking the appropriate area of the status bar

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

Changing options in this panel will affect *all* windows showing the same file, and not just the current window

The Options Current Modes *Input* Panel

This panel of the dialog shown by the **Options Current Modes** command controls how PFE acts when you type on the keyboard into the current window.

- Check the **Auto-indent new lines** box to have PFE indent each line to the same level as the previous one whenever you press the **Enter** key

- Check the **Strip trailing spaces when newline pressed** box to have PFE remove any trailing spaces to the right of a   line whenever you press the **Enter** key to start a new line

- Check the **Overtype existing text** box to have PFE replace characters already in the file with the ones you type, rather than have your typing inserted into what is already there.   Changing this option is equivalent to typing the **Ins** key on the keyboard, or double-clicking the left mouse button over the appropriate area of the status bar

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Options Current Modes *Language Awareness* Panel

This panel of the dialog shown by the **Options Current Modes** command controls the language set on the current window. This affects some aspects of how the text is displayed and how some commands operate.

- Select the language which you wish PFE to use from the **Language** list. Currently the languages supported are "**C**" and "**TeX**"; to turn off all language related options, set the value to "**(none)**"

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Options Current Modes *Print Defaults* Panel

This panel of the dialog shown by the **Options Current Modes** command control the default settings of various options related to printing from the current window. These values are used in the dialog shown by the **File Print** command.

- Check the **Print page headers** box to select that each page of printed output should have a header showing the name and modification date of the file, and a page number, by default

- Check the **Wrap long lines** box to select that lines too long to be printed on the page are folded and printed on as many lines as are required, rather than being truncated

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Options Current Modes *Saving* Panel

This panel of the dialog shown by the **Options Current Modes** command controls how PFE saves the file showing in the current window to disk.

- Check the **Use UNIX convention (LF) for end of line** box to have PFE save the file to disk using a single Line Feed (LF) character at the end of a line. This will produce a file that is in the format generally used by the UNIX Operating System; if the box is not checked, the file will be saved in the normal DOS format, with the end of each line marked with a Carriage Return (CR) and a Line Feed (LF) character pair

- Check the **Make backup of original file** box to have PFE make a copy of any existing file on disk with the same name before saving commences. You can configure system-wide backup options with the **Options Preferences** command

- Check the **Terminate disk file with Ctrl+Z** box if you want PFE to ensure that the file on disk ends with a **Ctrl+Z** character, adding one if necessary. This option is needed only if you need to write files in a format compatible with antiquated programs that expect files to use MS-DOS version 1 conventions.

- Check the **Do not add extra end-of-line after data** box if you want the data saved to the file to end with the last character typed, rather than with a CR-LF character pair marking the end of the last line

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

Changing options in this panel will affect *all* windows showing the same file, and not just the current window

The Options Current Modes *Text Colours* Panel

This panel of the dialog shown by the **Options Current Modes** command controls the text and background colours used in the current window.

- Click the **Use Colours set by control panel** button if you want the colours to be those you have defined with Control Panel

- Click the **Use specific colours** button if you want to specify your own colours. The list box below this button shows you the names of the various objects you can configure; you should take the following steps

    1 Double click the name of the object whose colour you want to change in the list box, or select it and click the **Change object colour** button

    2 This starts the standard colour selection dialog which allows you to choose the colour you want

    3 Repeat the steps above for the other objects you want to change. The samples at the top of the panel show you the effect of your choices

    Note that although the colour selection dialog allows you to choose dithered colours, PFE will not use these, but will instead use the nearest solid colour. The sample areas will show you the exact background colour you have specified at the left and right edges; the text and the background immediately behind the text will show the colours that will actually be used.

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Options Current Modes *Text Formatting* Panel

This panel of the dialog shown by the **Options Current Modes** command controls how text in the current window is formatted.

The **Tabs** area lets you control how TABs are inserted and appear

- Click the **Tab with tab characters** button to have PFE insert a single **TAB** character (ASCII code 9) when you press the **Tab** key. This mode of operation is often referred to as using *hard tabs*

- Click the **Tab with space characters** button to have PFE insert as many **SPACE** characters as are required to reach the next tab position when you press the **Tab** key. This mode of operation is often referred to as using *soft tabs*

- Set the width of a tab stop by entering the value in the **Tab stop every *n* characters** edit control. You can set any value between 2 and 16. Note that tabs can set only at regular intervals.

Note that changing how tabs are inserted affects only how the **Tab** key is handled *in the future*. Any **TAB** characters already in the file are not changed to spaces; nor are sequences of spaces condensed into **TAB** characters.

The **Text Wrapping** area controls aspects of how PFE wraps text when you reach the right hand margin

- Check the **Enabled** box to activate text wrapping

- Enter the column position of the right hand margin in the **At column** edit control. You can select any value greater than 16.

  Note that text wrapping occurs only when you type *at the end* of a line. PFE does not provide word-processor type text reflowing, where text in the current and following lines is repositioned wherever you type.

Options set in this panel are not applied until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes Dialog

This dialog lets you edit modes that apply to the mode group or window type selected by the parent dialog run by the **Options Default Modes** command.

The dialog box contains two major areas. To the left, there is a list box that allows you to select the category of modes you want to configure, and buttons that apply to the process of changing all the options. To the right is an area whose appearance will change depending on the category of modes you're working with.

Steps

1   Select the category of modes you want to configure from the list box on the left. The appearance of the panel on the right will change as you select the category, to show you the options that are relevant to it

2   In the panel on the right, set the various options as you want them. The individual panels are described in detail in the following help screens

3   If you want to apply the values you have changed and save them to the initialisation file, but carry on making further changes, click the **Apply Now** button.

4   If you want to apply the values you have changed, save them to the initialisation file, and close the dialog, click the **OK** button.

5   To close the dialog and abandon any changes you have made, click the **Cancel** button

Not all the various panels that make up the dialog are appropriate for all circumstances, and depending on that object you selected in the parent dialog, some may not appear. The full set of panels available are:

Display	What is displayed in windows showing the file by default, such as line numbers
File Properties	Various properties relating to the file, such as whether it is in read-only mode
Input	How PFE handles input to windows showing the file by default
Language Awareness	What language awareness should be used by default in windows
Loading	What actions PFE should take when loading a file whose filename matches a pattern defined in a mode group
Print Defaults	What default values should be assumed when printing the current window
Saving	What actions PFE should take saving a file whose filename matches a pattern defined in a mode group
Text Colours	What colours should be used by default in windows showing the file
Text Formatting	Text layout options such as tab size

The Edit Default Modes *Display* Panel

This panel of the Edit Default Modes dialog controls the default values of various options related to what is displayed in windows.

- Check the **Show line numbers** box to have PFE display the line number of each line in the window on the left side of the window

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *File Properties* Panel

This panel of the Edit Default Modes dialog controls the default values of some properties relating to the files displayed in windows.

The **Editing options** area contains options related to how you edit files:

- Check the **File is read only in all windows** box to have the file loaded in *read only* mode. You will not be able to change the file in any window showing it until you change the setting with the **Options Current Modes** command, or by double-clicking the appropriate area of the status bar

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Input* Panel

This panel of the <u>Edit Default Modes dialog</u> sets default values for various options affecting how PFE acts when you type on the keyboard.

- Check the **Auto-indent new lines** box to have PFE indent each line to the same level as the previous one whenever you press the **Enter** key

- Check the **Strip trailing spaces when newline pressed** box to have PFE remove any trailing spaces to the right of a line whenever you press the **Enter** key to start a new line

- Check the **Overtype existing text** box to have PFE replace characters already in the file with the ones you type, rather than have your typing inserted into what is already there.

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Language Awareness* Panel

This panel of the Edit Default Modes dialog controls the default values related to language awareness.

- Select the language which you wish PFE to use from the **Language** list. Currently the languages supported are "**C**" and "**TeX**"; to turn off all language related options, set the value to "**(none)**"

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Loading* Panel

This panel of the <u>Edit Default Modes dialog</u> controls the default values of options related to how PFE loads files.

- Check the **Discard any Ctrl+Z at end of data** box to have PFE automatically remove any **Ctrl+Z** character at the end of the file. This option does not affect the copy of the file on disk in any way. By default, PFE will load all characters in the file as they occur.

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Print Defaults* Panel

This panel of the Edit Default Modes dialog control the default settings of various options related to printing. These values are used in the dialog shown by the **File Print** command.

- Check the **Print page headers** box to select that each page of printed output should have a header showing the name and modification date of the file, and a page number, by default

- Check the **Wrap long lines** box to select that lines too long to be printed on the page are folded and printed on as many lines as are required, rather than being truncated

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Saving* Panel

This panel of the <u>Edit Default Modes dialog</u> controls the default values of options related to how PFE saves files to disk.

- Check the **Use UNIX convention (LF) for end of line** box to have PFE save the file to disk using a single Line Feed (LF) character at the end of a line. This will produce a file that is in the format generally used by the UNIX Operating System; if the box is not checked, the file will be saved in the normal DOS format, with the end of each line marked with a Carriage Return (CR) and a Line Feed (LF) character pair

- Check the **Make backup of original file** box to have PFE make a copy of any existing file on disk with the same name before saving commences. You can configure system-wide backup options with the **<u>Options Preferences</u>** command

- Check the **Terminate disk file with Ctrl+Z** box if you want PFE to ensure that the file on disk ends with a **Ctrl+Z** character, adding one if necessary. This option is needed only if you need to write files in a format compatible with antiquated programs that expect files to use MS-DOS version 1 conventions.

- Check the **Do not add extra end-of-line after data** box if you want the data saved to the file to end with the last character typed, rather than with a CR-LF character pair marking the end of the last line

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Text Colours* Panel

This panel of the Edit Default Modes dialog controls the default values of the text and background colours used when showing a file.

- Click the **Use Colours set by control panel** button if you want the colours to be those you have defined with Control Panel

- Click the **Use specific colours** button if you want to specify your own colours. The list box below this button shows you the names of the various objects you can configure; you should take the following steps

    1   Double click the name of the object whose colour you want to change in the list box, or select it and click the **Change object colour** button

    2   This starts the standard colour selection dialog which allows you to choose the colour you want

    3   Repeat the steps above for the other objects you want to change. The samples at the top of the panel show you the effect of your choices

        Note that although the colour selection dialog allows you to choose dithered colours, PFE will not use these, but will instead use the nearest solid colour. The sample areas will show you the exact background colour you have specified at the left and right edges; the text and the background immediately behind the text will show the colours that will actually be used.

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Default Modes *Text Formatting* Panel

This panel of the Edit Default Modes dialog controls the default values of various options related to how text in windows is formatted.

The **Tabs** area lets you control how tabs are inserted and appear

- Click the **Tab with tab characters** button to have PFE insert a single **TAB** character (ASCII code 9) when you press the **Tab** key. This mode of operation is often referred to as using *hard tabs*

- Click the **Tab with space characters** button to have PFE insert as many **SPACE** characters as are required to reach the next tab position when you press the **Tab** key. This mode of operation is often referred to as using *soft tabs*

- Set the width of a tab stop by entering the value in the **Tab stop every *n* characters** edit control. You can set any value between 2 and 16. Note that tabs can set only at regular intervals.

The **Text Wrapping** area controls aspects of how PFE wraps text when you reach the right hand margin

- Check the **Enabled** box to activate text wrapping

- Enter the column position of the right hand margin in the **At column** edit control. You can select any value greater than 16.

  Note that text wrapping occurs only when you type *at the end* of a line. PFE does not provide word-processor type text reflowing, where text in the current and following lines is repositioned wherever you type.

Options set in this panel are not made permanent until you click the **Apply Now** or **OK** buttons. To cancel any changes you have made, click the **Cancel** button.

The Edit Goto Line Dialog

This dialog lets you move to the start of an arbitrary line in the current window. You can move either to an absolute line number, or move up or down in the window by a given number of lines.

Steps

1 Enter the place within the file you want to move to in the **Destination** box. The dialog shows you the number of the last line in the current file, and of the line the caret is now in, for information.

   You can type a simple line number, or give a more complex <u>file position specification</u>

2 If you want to highlight all the text from where the caret now is and the start of the target line, check the **Extend Selection** box

3 Click the **OK** button to go to the target line

The Edit Text Insert ASCII Code Dialog

This dialog lets you insert a character specified by its ASCII code number into the current file at the position of the caret

Steps

1 Specify the character you wish to insert.

To insert a standard control character such as **ESC** or **Ctrl+Z**, click on the **Control Char** button and select the mnemonic of the control character you want from the list to its right. If you wish to set the top bit (bit 7) of the character, check the **Top Bit Set** box

To insert a character with an arbitrary ASCII code, click on the **ASCII code** button and type the code as a decimal number in the edit control to its right. You can enter any number between 1 and 255.

2 Click the **OK** button to insert the character with the specified ASCII code at the position of the caret

The File Print Dialog

This dialog lets you print some or all of the current file on your currently selected printer

Steps

    1 To use a different printer from the one shown at the top of the dialog box; to change the printer font; or to change the page margins, click the **Setup Printer** button

    2 Decide what part of the file you want to print by using the **Print** area:

        Click **Whole File** (the default) to print everything

        Click **Selected Text** to print exactly the text that is highlighted. This option will not be available if you don't have any text highlighted

        Click **Line Range**, and fill in the start and end line numbers, to print a range of lines. You can use the words **"start"** and **"end"** to represent the first and last lines of the file

    3 If you want to use different print options from those set by the <u>window modes</u> of the current window, click the boxes in the **Options** area:

        If **Number Lines** is checked, lines are printed with numbering

        If **Wrap Long Lines** is checked, lines too long to fit the page are folded rather than being truncated

        If **Page Headers** is checked, each page starts with a header giving the file name and other information

The File Print Setup Dialog

This dialog lets you set up details of the printer that PFE is to use, and to run the printer's own setup dialog. Settings you make in this dialog are recorded and become the defaults.

You can set different values for each of the printers configured on your system.

Steps

1 If you want to use a different printer to the one that is highlighted in the **Available Printers** list, scroll the list and click the left button on the printer name

2 To change the margins used on each page, check or uncheck the boxes in the **Print Options** area. All margins are 0.5 inches wide

3 To run the highlighted printer's own setup dialog, click the **Setup** dialog.

4 To change the printer font that PFE will use for the highlighted printer, click the **Font** button

If you run the printer's own setup dialog *after* selecting a font, you may find that the font is no longer available; some printers offer different fonts in different operating modes.

The Template Create File Dialog

This dialog allows you to create a new, empty template file that you can use to store templates.

Template files have a file type of "**.tpl**"

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that you want to create the template file in

2 Type the name you want to call the template file, or select an existing name from the **Files** list. If you do specify an existing file, PFE will prompt you to confirm that you really want to overwrite it.

3 Click the **OK** box to create the file

If you prefer, you can simply type the name of the file to be attached directly into the **File Name** edit control rather than use the browse facilities.

About Templates

The Template Edit Dialog

This dialog allows you to edit a template contained in a template file. You must first have attached the template file, either automatically or with the **Template Attach File** command.

Steps

1. Select the template file containing the template you want to edit from the **Template Files** list. As you select a file, the list of templates it holds will appear in the **Template Names** list.

2. Select the name of the template to edit from the **Template Names** list

3. Click the **OK** button to edit the template and close the dialog

PFE opens a new edit window and copies the contents of the template into it for you to change as you require.

When you've finished amending the template, you can update the in-memory copy of the template file with the **Template Store** command, then save the changed template file to disk with the **Template Save File** command.

About Templates

The Template Store As Dialog

This dialog allows you to store a template that you're editing into any attached template file, giving it a new name in the process.

Steps

    1 Select the template file into which you want to store the template from the **Template Files** list on the left. The list of templates already in this template file will be updated.

    2 Type the name you want to give the template into the **Template Name** edit control on the right, or select the name of an existing template. A template name can be up to 16 characters long, and may contain spaces.

    3 Click the **OK** button to store the template in the in-memory copy of the template file, under the new name, and close the dialog.

       If the name you're using is that of an existing template, PFE will ask you to confirm that you really want to overwrite it.

Note that this operation changes only the in-memory copy of the template file. The file on disk will not be updated until you use the   **Template Save File** command.

      About Templates

The Template Delete Dialog

This dialog allows you to delete one or more templates from a template file.   You must first have attached the template file, either automatically or with the **Template Attach File** command.

Steps

1 Select the template file containing the templates you want to delete from the **Template Files** list. As you select a file, the list of templates it holds will appear in the **Template Names** list.

2 Select the name of the templates to delete from the **Template Names** list. You can use **Ctrl** to add names to those selected, and **Shift** to select a range of names. You can also click and drag over a range of names.

3 Click the **Delete** button to delete the templates from the in-memory copy of the template file and close the dialog. To update the template file on disk, you need then to use the **Template Save File** command.

About Templates

The Template Detach File Dialog

This dialog allows you to detach one or more attached template files, freeing main memory, when you no longer require them

Steps

1 Select one or more template files from the list of those attached. You can use the **Ctrl** key to add files to the selection; or the **Shift** key to extend a selection. You can also select multiple files by clicking and dragging.

2 Click the **OK** button to detach the template files and close the dialog. If you've changed any of the templates contained in a template file, you'll be asked if you want to save the changes to disk or to discard them.

Once you've detached a template file, you can't use the templates it contains until you attach it again.

You can't detach a template file if you're currently editing any template that it contains.

About Templates

The Template Save File Dialog

This dialog allows you to save changes that you've made to one or more attached template files to disk, so that they become permanent.

Steps

1 Select the template files you want to save from the list. You can use **Ctrl** to add filenames to the selection, and **Shift** to select a range. You can also click and drag over a range of names.

2 Click the **Save** button to save the in-memory copies of the selected template files to disk and close the dialog. The previous contents of the template files will be overwritten

About Templates

The Template Insert   Dialog

This dialog allows you to insert a template from any attached template file into the current window.

Steps

1. Select the template file containing the template you want to insert from the **Template Files** list. As you select a file, the list of templates it holds will appear in the **Template Names** list.

2. Select the name of the template to insert from the **Template Names** list

3. Click the **OK** button to insert the template and close the dialog

The template will be inserted into the current window at the position of the caret. The caret will be positioned at the start of the inserted template data after this operation

<u>About Templates</u>

The Key Mapping Save As Dialog

This dialog allows you to save changes that you've made to the current <u>key mappings</u> into a file of your choice

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that you want to save the key mapping file into

2 Either type the name of the file you want to save the data into in the **File Name** edit control, or select the name of an existing file you want to overwrite from the list box. Key mapping files should normally be given a file type of "**.key**"

3 Click the **OK** box to save the current key mappings to the file. If you've specified the name of a file that already exists, PFE will ask you to confirm that you really intend to overwrite it.

The Key Mapping Load Dialog

This dialog allows you load a set of <u>key mappings</u> from a file of your choice

Steps

   1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that the key mapping file is in

   2 Select the file you want to open in the **File Name** list box. Normally, key mapping files have a file type of "**.key**"

   3 Click the **OK** box to load the key mappings contained in the file

If you prefer, you can simply type the name of the file containing the key mappings directly into the **File Name** edit control rather than use the browse facilities.

The File Open Dialog

This dialog allows you to open one or more files that already exist.

Steps

    1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that the files are in

    2 Select the files you want to open in the **File Name** list box. You can use the **Ctrl** key to add filenames to those selected, and the **Shift** key to select a range of filenames. You can also click and drag across a sequential series of names.

    3 If you want to open the files in *read only* mode, check the **Read Only** box. This will affect *all* the files you open in this operation.

    4 Click the **OK** box to open the files

If you prefer, you can simply type the name of the file or files to open directly into the **File Name** edit control rather than use the browse facilities.

You can repeat steps **1** and **2** to open files from several drives and directories in the same operation.

The **List Files of Type** list at lower left allows you to restrict the files shown in the list to those matching specific filename patterns. By default, there is one entry - "***.***" - in this list, so you'll see all the files in each directory. You can <u>configure</u> the list to match how you want to work with the **Options Preferences** command.

Whether you close the dialog with the **OK** or the **Cancel** buttons, PFE will make its current working directory the one that the dialog is showing.

If you're already editing one of the files you specify, PFE won't load a new copy from disk, but will simply activate a window showing the file.

The File Save As Dialog

This dialog allows you to save the file in the current window to disk, in a file whose name you specify. The name you give for the disk file is then associated permanently with it the current window and all others showing the same file,

Steps

1   Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that you want to save the file to

2   Either type the name of the file you want to save the data into in the **File Name** edit control, or select the name of an existing file you want to overwrite from the list box.

3   If you want to save the file in Unix format, check the **Save In Unix Format** box. For normal DOS format, make sure that the box is unchecked.

    The setting you use here becomes the default for the next time you save this specific file.

4   Click the **OK** box to save the data to the file. If you've specified the name of a file that already exists, PFE will ask you to confirm that you really intend to overwrite it.

The **List Files of Type** list at lower left allows you to restrict the files shown in the list to those matching specific filename patterns. By default, there is one entry - "***.***" - in this list, so you'll see all the files in each directory. You can configure the list to match how you want to work by editing the initialisation file.

After you've saved the current file with this dialog, PFE will associate the name of the disk file with all the windows showing it. The **File Save** command will then write to this file automatically.

The File View Dialog

This dialog allows you to open one or more files that already exist in *read only* mode. Its action is exactly as if you used the **File Open** command and checked the **Read Only** box.

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that the files are in

2 Select the files you want to open in the **File Name** list box. You can use the **Ctrl** key to add filenames to those selected, and the **Shift** key to select a range of filenames. You can also click and drag across a sequential series of names.

3 Click the **OK** box to open the files

If you prefer, you can simply type the name of the file or files to open directly into the **File Name** edit control rather than use the browse facilities.

You can repeat steps **1** and **2** to open files from several drives and directories in the same operation.

The **List Files of Type** list at lower left allows you to restrict the files shown in the list to those matching specific filename patterns. By default, there is one entry - "***.***" - in this list, so you'll see all the files in each directory. You can configure the list to match how you want to work with the **Options Preferences** command.

Whether you close the dialog with the **OK** or the **Cancel** buttons, PFE will make its current working directory the one that the dialog is showing.

If you're already editing one of the files you specify, PFE won't load a new copy from disk, but will simply activate a window showing the file.

The File Name Dialog

This dialog allows you to change the file name associated with the file in the current window. The **File Save** command will then write to this file automatically.

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that you want to use in the file name

2 Either type the name of the file you want to save the data into in the **File Name** edit control, or select the name of an existing file from the list box.

3 Click the **OK** box to associate the new file name with the file in the current window

If you prefer, you can simply type the name of the file directly into the **File Name** edit control rather than use the browse facilities.

The **List Files of Type** list at lower left allows you to restrict the files shown in the list to those matching specific filename patterns. By default, there is one entry - **"*.*"** - in this list, so you'll see all the files in each directory. You can configure the list to match how you want to work with the **Options Preferences** command.

The File Write Dialog

This dialog allows you to save the file in the current window to disk, in a file whose name you specify. Unlike the **File Save As** command, the filename associated with the window is *not* changed.

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that you want to write the file to

2 Either type the name of the file you want to save the data into in the **File Name** edit control, or select the name of an existing file you want to overwrite from the list box.

3 Click the **OK** box to save the data to the file. If you've specified the name of a file that already exists, PFE will ask you to confirm that you really intend to overwrite it.

The **List Files of Type** list at lower left allows you to restrict the files shown in the list to those matching specific filename patterns. By default, there is one entry - "***.***" - in this list, so you'll see all the files in each directory. You can configure the list to match how you want to work with the **Options Preferences** command.

The File Insert Dialog

This dialog allows you to insert the entire contents of an existing file into the current window at the position of the caret.

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that the file is in

2 Select the file you want to open in the **File Name** list box

3 Click the **OK** box to insert the entire contents of the file

If you prefer, you can simply type the name of the file to be inserted directly into the **File Name** edit control rather than use the browse facilities.

The **List Files of Type** list at lower left allows you to restrict the files shown in the list to those matching specific filename patterns. By default, there is one entry - **"*.*"** - in this list, so you'll see all the files in each directory. You can <u>configure</u> the list to match how you want to work with the **<u>Options Preferences</u>** command.

The Edit Find Dialog

This dialog allows you to find occurrences of text strings.   Unlike most dialogs, you can continue to work in PFE while it is visible, so you can switch between your windows to search in several of them in one operation.

Steps

1 Type the text you want to search for in the **Find What** edit control. A special <u>notation</u> allows you to search for characters like "end of line" that you can't type. The notation uses the "\" character, so you must type this as "\\".

If the current window contained a highlighted selection when you started the dialog, the **Find What** edit control will show the selected text, allowing you quickly to search for further occurrences. PFE also records the last 8 strings you searched for, and you can repeat the search for one of these by clicking on the drop-down button to the right of the edit control and selecting the string from the list. The string you searched for last time will always be at the head of the list.

2 In the **Direction** area, set the direction of search. You can search *up* from the caret's current position to the start of the file, or *down* from the caret's position to the end of the file

3 Set up the search options you want to apply:

If the **Match Case** box is checked, PFE will match what you type exactly. If it's unchecked, upper-case and lower-case letters are taken as the same

If the **Select Matching Text** box is checked, PFE will highlight any matching text it finds. If it's unchecked, the text won't be highlighted

If the **Extend Selection** box is checked, PFE will highlight all the text from where the caret is now to the end of the matching text. You can only check this box if you've also checked the **Select Matching Text** box.

4 Click the **Find Next** button to perform the search. You can repeat this as often as you want.

If you want to search in some other window, simply activate it while this dialog is still on screen and click the **Find Next** button again.

5 To close the dialog, click the **Cancel** button.

After you've performed a search, you can repeat it very quickly with the **<u>Edit Repeat Last Find</u>** command, which will make the search without showing this dialog.

By default the strings saved in the **Find What** list are not recorded for subsequent edit sessions. However, you can request that PFE does record them with the **<u>Options Preferences</u>** command.

The Edit Replace Dialog

This dialog allows you to find occurrences of a text string and replace them with another.   Unlike most dialogs, you can continue to work in PFE while it is visible, so you can switch between your windows to make changes in several of them in one operation.

The search is always made from the current position of the caret down to the end of the file.

Steps

1   Type the text you want to search for in the **Find What** edit control. A special <u>notation</u> allows you to search for characters like "end of line" that you can't type. The notation uses the "\" character, so you must type this as "\\".

If the current window contained a highlighted selection when you started the dialog, the **Find What** edit control will show the selected text, allowing you quickly to search for further occurrences. PFE also records the last 8 strings you searched for, and you can repeat the search for one of these by clicking on the drop-down button to the right of the edit control and selecting the string from the list. The string you searched for last time will always be at the head of the list.

2   Type the text you want to replace it with in the **Replace With** edit control. The same <u>notation</u> is used for characters you can't type.

PFE also records the last 8 strings you used as replacement strings, and you can re-use one of these by clicking on the drop-down button to the right of the edit control and selecting the string from the list. The string you used last time will always be at the head of the list.

To simply delete occurrences of the search string, leave this edit control blank.

3   Set up the search options you want to apply:

If the **Match Case** box is checked, PFE will match what you type exactly. If it's unchecked, upper-case and lower-case letters are taken as the same

4   Select an operation you want to perform:

Clicking the **Find Next** button moves to the next occurrence of the search string

Clicking the **Replace** button replaces the current match (or the first match it finds after the caret), and moves to the next

Clicking the **Replace All** button replaces all matches automatically

5   To close the dialog, click the **Cancel** button.

You can repeat step **4** as often as you wish. To work in another window, simply activate it and choose an action from step **4**.

Clicking the **Undo Last** button will reverse the effect of the last replacement made in the current window. To undo the effects of more than one replacement, use the **<u>Edit Undo</u>** menu command or the undo <u>toolbar</u> button.

After you've performed a replace operation, you can repeat it very quickly with the **<u>Edit Repeat Last Replace</u>** command, which will make the replacement without showing this dialog.

By default the strings saved in the **Find What** and **Replace With** lists are not recorded for subsequent

edit sessions. However, you can request that PFE does record them with the **Options Preferences** command.

The Template Attach File Dialog

This dialog allows you to *attach* a template file, making the templates that it contains available for use.

Template files have a file type of "**.tpl**"

Steps

1 Use the **Drive** and **Directory** list boxes to select the disk drive and the directory that the template file is in

2 Select the file you want to open in the **File Name** list box

3 Click the **OK** box to attach the file

If you prefer, you can simply type the name of the file to be attached directly into the **File Name** edit control rather than use the browse facilities.

About Templates

The Execute DOS Command To Window Dialog

This dialog allows you to run a DOS program, such as a compiler, or an internal command such as **dir**, and capture the <u>output</u> into an edit window.

You can run only one DOS program at a time with this dialog.

You can't start a Windows application with this dialog; for that, use the **<u>Execute Launch Application</u>** command.

Steps

1 Type the DOS command line you want to execute into the **Command** edit control. You can automatically <u>substitute</u> parts of the name of the file showing in the current window in the command line if you wish. Such substitution points are indicated by a "**%**" character, so if you wish to include a "**%**" in the command line unaltered, type it as "**%%**"

You can also select one of the previous command lines used from the drop down list box.

To browse your directories to find the program you want to run, click the **Browse** button.

2 Type the path of the working directory you want the command to start in into the **Working Directory** edit control. If you leave this blank, the command starts in PFE's current working directory, which is shown at the top of the dialog.

3 Set the various options you want to apply:

If the **Beep When Done** box is checked, PFE sounds the standard system beep when the DOS program finishes.

If the **Minimize Editor** box is checked, PFE will make itself into an icon before running the DOS program

If the **Save Changed Files** box is checked, PFE will see if any of the files you're editing have changed, and will give you the opportunity to save the changes before running the DOS program. If you agree to save the changes, *all* the files you've altered will be written to disk.

If the **Reuse Output Window** box is checked, PFE will place the DOS program's output into the window used the last time you used this dialog. If not, a new output window will be created.

If the **Show End Of Output** box is checked, PFE will automatically scroll the window showing the output to show you the end of it rather than the start.

4 Click the **OK** button to run the DOS program.

The command line, working directory path and options are recorded, and become the default for the next time you use this dialog.

Under Windows 3, to change the properties of the DOS window used to run the program, use the standard **PIF Editor** to modify the PIF file **$pfedos.pif**

To repeat the operation exactly, use the **<u>Execute Repeat DOS Command to Window</u>** command.

The Execute Launch Application Dialog

This dialog allows you to launch a Windows application, or a DOS program, to run independently of PFE.

You can't capture the output of a DOS program launched with this dialog. For that, use the **Execute DOS Command To Window** command.

Steps

1 Type the command line for the application you want to launch you want to execute into the **Command** edit control, or select one of the previous command lines you used from the drop-down list box

   To browse your directories to find the application, click the **Browse** button.

2 Type the path of the working directory you want the application to start in into the **Working Directory** edit control. If you leave this blank, the application starts in PFE's current working directory, which is shown at the top of the dialog.

3 Set the various options you want to apply:

   If the **Minimize Editor** box is checked, PFE will make itself into an icon before running the DOS program

   If the **Save Changed Files** box is checked, PFE will see if any of the files you're editing have changed, and will give you the opportunity to save the changes before launching the application. If you agree to save the changes, *all* the files you've altered will be written to disk.

4 Click the **OK** button to launch the application.

The command line, working directory path and options are recorded, and become the default for the next time you use this dialog.

The Execute Launch Windows Tool Dialog

This dialog lets you launch one of the Windows Tools that you've configures in the list of Windows Tools with the **Execute Configure Windows Tools** command.

Steps

1 Select the name of the tool that you want to launch from the list at the lower left of the dialog. The command line and options that you defined for the tool will be shown.

2 If you wish, modify the details of the command line shown in the **Command Line** edit control. .

3 If you wish, modify the options that apply in the **Options** area:

If the **Minimize Editor** box is checked, PFE will make itself into an icon when you launch the tool

If the **Save Changed Files** box is checked, PFE will see if you have changed any of the files you're editing before launching the tool, and give you a chance to save them.

4 Enter the path name of the directory that you wish the tool to run in into the **Working Directory** edit control. If you leave this field blank, the tool will be run in PFE's current working directory, which is shown at the top of the dialog.

5 Click the **OK** button to launch the tool.

The values you set in this dialog are recorded, and will be used as the defaults when next you launch the same tool. The **Reset Command Line** button can be used at any time to restore the command line to the one you defined originally.

The Execute Configure Windows Tools Dialog

This dialog lets you add entries to the list of Windows tools, and to delete or change existing entries.

About Windows Tools

ADDING A NEW ENTRY

To add a new entry to the list of Windows tools, do this:

Steps

1   Type the name you want to use for the tool in the **Tool** list at the top of the dialog. A tool name can be up to 32 characters long, and can contain spaces.

2   Type the command line you want to associate with the tool name in the **Command Line** edit control. You can browse your disks and directories to find the name of the executable module by clicking the **Browse** button.

3   Set the default options you want to apply to the tool name in the **Options** area:

   If the **Minimize Editor** box is checked, PFE will make itself into an icon when you launch the tool

   If the **Save Changed Files** box is checked, PFE will see if you have changed any of the files you're editing before launching the tool, and give you a chance to save them.

4   Click the **Add** button to add the tool name and details to the list.

5   If you have more tools to add, return to step **1**; or follow the procedures below to change or delete other entries. When you've finished, click the **Save** button if you want to record the amended list permanently.

DELETING AN EXISTING ENTRY

To delete an existing entry from the Windows Tools lists, do this:

Steps

1   Select the name of the tool you want to delete in the **Tool** list at the top of the dialog.

2   Click the **Delete** button to delete the entry

3   If you have more tools to delete, return to step **1**; or follow the other procedures given here to add or change other entries. When you've finished, click the **Save** button if you want to record the amended list permanently.

CHANGING AN EXISTING ENTRY

To alter some or all of the details of an existing entry in the Windows Tools list, do this:

1   Select the name of the tool whose details you want to amend in the **Tool** list at the top of the

dialog.

2 If you want, modify the command line you want to associate with the tool name in the **Command Line** edit control. You can browse your disks and directories to find the name of the executable module by clicking the **Browse** button.

3 If you want, modify the default options you want to apply to the tool name in the **Options** area:

If the **Minimize Editor** box is checked, PFE will make itself into an icon when you launch the tool

If the **Save Changed Files** box is checked, PFE will see if you have changed any of the files you're editing before launching the tool, and give you a chance to save them.

4 Click the **Change** button to change the details associated with the tool name to those now set in the dialog.

5 If you have more tools to change, return to step **1**; or follow the procedures above to add or delete other entries. When you've finished, click the **Save** button if you want to record the amended list permanently.

The Options Key Mapping Dialog

This dialog allows you to alter the functions that are invoked when you press certain keys. The dialog allows you map new keys to functions; to quickly change a key mapping; and to configure prefix keys.

To map a key to a function, you can begin either with the key code, or with the name of the function. If you want to map a specific key to a function, it's easiest to begin by specifying the key; if you want to change the keys that map to a given function, it's best to start with the function name. However, you can use either method as you prefer.

TO MAP A KEY TO A FUNCTION

If you want to map a specific key to a function, it's best to start by specifying the key. Take the steps given here:

1 If the key you want to map is a prefix key, select the prefix key from the drop-down list at the top of the dialog. If the prefix key you want to use does not appear in the list, click the **Configure** button to run the prefix key configuration dialog. You can use **Esc** or any control key from **Ctrl+A** to **Ctrl+Z** as a prefix key.

2 Specify the second (or only) key to press from the controls in the next row of the dialog

Check the **Alt** box to specify that the **Alt** key must be held down

Check the **Ctrl** box to specify that the **Control** key must be held down

Check the **Shift** box to specify that the **Shift** key must be held down

Then select the key itself from the drop-down list on the right.

If the key can be mapped to a function, the name of the key will appear in the box below as confirmation. PFE will also indicate whether or not the key is already mapped to a function.

3 In the **Functions** section at the bottom of the dialog, select the name of the function you want to map the key to from the drop-down list. PFE will show you the names of any keys already mapping to this function in the list below the name

4 If you want the key name to appear on the menu associated with this function, check the **Show this mapping on menu** box; uncheck it if you do not want the key name to appear. This option won't be available if the function you've chosen has no corresponding menu item

5 Click the **Add/Change** button at the top right of the dialog to map the selected key to the selected function

TO UNMAP A   KEY FROM A FUNCTION

To unmap a key from a function, you can if you like use the steps detailed above and start by specifying the key. However, it's probably easier if you use the steps below and start with the function name.

1 In the **Functions** section of the dialog, select the name of the function from the drop-down list. PFE will show the names of the keys currently mapping to this function in the list below the function name

2 Select the key whose mapping you want to delete by double-clicking the left mouse button on the name. The key details will appear at the top of the dialog.

3 If you simply want to unmap the key so it does nothing, click the **Delete** button.

If you want to map the key to a *different* function, select the new function name from the drop-down list in the **Commands** section of the dialog, and click the **Add/Change** button.

SAVING CHANGES

Changes you make in this dialog apply only to the current PFE session unless you save them.

To save the changes so that they become the default key mapping that PFE will use when it starts, click the **Save** button. To avoid the risk of accidental change, if you specified a key map file when you started PFE with a "**/k**" command line option, the **Save** button will not be available. In this case, you must always use the **Save As** button to save changes.

To save the changes into a key map file, so that you can manually load and apply them when you wish, click the **Save As** button. This starts a sub-dialog that lets you specify the name of a file into which the key mappings can be saved.

LOADING A SET OF KEY MAPPINGS

To load a set of key mappings that you've saved to a file with the **Save As** button, click the **Load** button. This starts a sub-dialog that lets you specify the name of a file containing mappings, which PFE will load and apply.

The Options Prefix Keys Dialog

This dialog lets you define which keys are used as *prefix keys*, or keys which start two-character command sequences. You can use the **Esc** key, or any control key from **Ctrl+A** to **Ctrl+Z** as a prefix key

About Key Mapping

ACTIVATING KEYS AS PREFIXES

To make keys that are currently not prefixes act as prefixes, select the key names from the **Inactive prefix keys** list on the left of the dialog, and click the **Activate** button. The names will transfer to the **Active prefix keys** list on the right.

Note that if you make a key an active prefix, any mappings you've set up for it *by itself* will be suspended.

DEACTIVATING KEYS AS PREFIXES

To make keys that are currently prefixes act as ordinary keys, select the key names from the **Active prefix keys** list on the right of the dialog, and click the **Deactivate** button. The names will transfer to the **Inactive prefix keys** list on the left.

The Screen Font Dialog

This dialog allows you to choose the screen font that PFE will use to display text in all its edit windows. You can choose any fixed-pitch font that's installed in your system.

Steps

1 Select the name of the font you want to use from the **Font** list on the left.

2 If you want some other style than plain text, such as italic or bold, select this from the **Font Style** list in the centre of the dialog.

3 Select the size of the font, in points, from the **Size** list at the right. If the font you've selected is a *scalable* font - such as a TrueType font or one generated by Adobe Type Manager - you can choose any size between 6 and 24 points. Other fonts may offer only certain sizes.

4 Click the **OK** button to select the font and close the dialog.

PFE will use the font details you've chosen for all windows. The details are recorded and become the defaults.

The Options Preferences Dialog

This dialog allows you to configure the way in which you wish PFE to operate.

The dialog box contains two major areas. To the left, there is a list box that allows you to select the category of items you want to configure, and buttons that apply to the process of changing all the options. To the right is an area whose appearance will change depending on the category of items you're working with.

Steps

1. Select the category of items you want to configure from the list box on the left. The appearance of the panel on the right will change as you select the category, to show you the options that are relevant to it

2. In the panel on the right, set the various options as you want them. The individual panels are described in detail in the following help screens

3. If you want to apply the values you have changed and save them to the underlined initialisation file, but carry on making further changes, click the **Apply Now** button.

4. If you want to apply the values you have changed, save them to the underlined initialisation file, and close the dialog, click the **OK** button.

5. To close the dialog and abandon any changes you have made, click the **Cancel** button

Note that a small number of options cannot be changed while PFE is running. The new values you set will be recorded, but will not take effect until you exit PFE and restart it. If you alter any of these values, PFE will warn you about them when you click either the **Apply Now** or **OK** buttons.

The various panels that appear in the area on the right of the dialog are:

Backup	Options related to backing up files when they are saved
Context Help	Options related to providing help on items within files
DOS Shell	Configures the DOS shell that PFE can start for you
Editing Files	General editing options
File Filters	The filters shown in the various dialogs that select files for opening
General Options	General control options
Key Mapping	Options related to mapping keys to functions
MRU List	Configures the list of Most-Recently-Used files that appears on the **File** menu
Saving Files	Options related to saving files to disk
Screen Font	Options that configure the screen font that PFE uses
Starting PFE	Options that configure what PFE does when it starts
Status Bar	Options that control the status bar
System Colours	The colours of items such as the application workspace
Templates	Options related to templates
Tool Bar	Options related to the toolbar
User Help Files	Options relating to the user-configurable items added to the **Help** menu
Window Control	Options that control the initial state of edit windows
Windows Managers	Options relating to the various *manager* items on the **Execute** menu

The Options Preferences *Backup* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Backup** option from the list box on the left.

It allows you to configure various options that relate to how PFE keeps backup copies of existing files when you save the changes you have made in an edit window.

- Click the **Never** button if you do not want PFE to make backup copies of files under any circumstances. The various other configuration and saving dialogs will not allow you to select backup-related items in this case.

- Click the **Into file with extension** button if you want PFE to make a backup copy into the same directory as the original file, but with a special file extension. You can choose the file extension you want PFE to use for backup files - by default, it will be "**.$$$**"

- Click the **Into sub-directory** button if you want PFE to make a backup copy into a sub-directory of the directory holding the original file, and using the same file name as the original. You can choose the sub-directory name (which must be in *8.3* format) or use the default of "**$PFEBK**". Note that PFE will create the sub-directory for you if it does not already exist.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Context Help* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Context Help** option from the list box on the left.

It allows you to configure various options that relate to how PFE provides you with context specific help when you double click the right mouse button on a word in an edit window or use the **Help Context Help** command.

The **Context Help File** area allows you to specify the Windows help file that will be opened to provide context-specific help.

-   Click the **Use first user help file** button to use the first help file specified in the list maintained by the **User Help Files** panel of this dialog

-   Click the **Use this file** button if you want to specify a specific help file. You can enter the name of an existing Windows help file, or browse for one by clicking the "**...**" button to the right of the edit control. Note that any file name you specify will be converted to an absolute pathname.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Browse For Context Help File Dialog

This dialog allows you to browse your disks and directories to find a Windows help file to be used to supply context help when you use the **Help Context Help** command or double-click the right mouse button on a word in an edit window.

Steps

1   Use the **Drives** and **Directories** lists to move around your files until you find the directory containing the Windows help file.

2   Select the name of the file from the **File Name** list on the left.

3   Click the **OK** box to close the dialog. The full path name of the file you've selected will appear in the **Path** edit control of the dialog panel.

The Options Preferences *DOS Shell* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **DOS Shell** option from the list box on the left.

It allows you to configure the DOS Shell that PFE starts when you use the **Execute DOS Prompt** command or click the DOS Shell tool bar button

- Click the **Use COMSPEC** button to have PFE determine the pathname of the command processor from the contents of the **COMSPEC** environment variable. This is the default situation, and you will probably not need to change it unless you are using a command processor other than **command.com** for the 16-bit edition or **cmd.exe** for the 32-bit edition. Alternative command processors will normally ensure that the **COMSPEC** environment variable is set to refer to them. With this option set, PFE runs the command processor with no command line arguments

- Click the **Use this command line** button if you want to specify the exact command line to be run. You can type in the command line you want to be executed -   clicking the "**...**" button starts a dialog that lets you browse for the name of the command processor itself

Changes you make with this panel will not be recorded or actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Editing Files* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Editing Files** option from the list box on the left.

It allows you to configure various options that relate to how PFE edits files.

- Check the **Remove highlight after copy to clipboard** box if you want PFE to remove the highlight from selected text after you use the **Edit Copy** command to copy it to the clipboard. The default is to leave the text highlighted.

- Check the **Allow drag and drop** editing button if you want to be able to move text around in an edit window by dragging it with the mouse and dropping it where you want it to be. The default is to have drag and drop editing active.

- Check the **Default drag and drop action to move** box if you want the drag and drop editing action to delete the text from where it is originally if you simply hold the mouse button down while dragging. With the box unchecked, you will need to hold the **Control** key down to delete the original copy

- Check the **Beep on all errors** box if you want audio feedback on all errors. With the box unchecked, only errors that do not give you any other indication will make a sound. The default is to beep always.

- Check the **Highlight result of Edit Find action** box if you want the text matched by the **Edit Find** command to be highlighted. The default is to have highlighting turned on.

- Check the **Scroll to follow vertical thumbtack** box if you want PFE to scroll the text in a window to follow the movement of the thumbtack in its vertical scroll bar. The default is to have scrolling turned on.

- Check the **Detect file format when opening file** box if you want PFE to attempt to detect whether a file is in DOS or UNIX formats when it is loaded. The default is to have detection turned on.

- Check the **Use block caret for insert mode** box if you want PFE to show the caret as a solid block when you are inserting text, and as a thin vertical line when you are overwriting existing text. With the box unchecked (the default) the caret is a box for overwriting and a line for inserting.

- The **Maximum undo actions recorded** control lets you configure the amount of undo action recorded for each file being edited. You can set the value to a number between 8 and 128; larger numbers will consume more memory and will cause a small performance degradation in some operations. Note that an undo action is a *sequence* of keystrokes and not a single one; thus an unbroken typing sequence counts as *one* action, rather than as the number of characters typed.

  You will need to restart PFE to action any change to this value.

Changes you make with this panel will not be recorded or actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *File Filters* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **File Filters** option from the list box on the left.

It allows you to configure the list of filters used to select files in the standard file-open dialogs used by the **File Open** and similar commands.

The dialog panel allows you to specify new filters, edit existing ones, and change the ordering of how the filters appear in the dialogs.

Defining a new filter

To define a new file filter, take the following steps:

1 In the **Filter String** edit control at the bottom of the dialog, type the filter string you require. This is a list of file names, which will normally include DOS wildcard characters. For example, you might type the string

```
 *.c , *.h , *.rc
```

to set up a filter that selects files with extensions "**.c**", "**.h**" and "**.rc**".

You can specify up to 16 file filters in each string, and can separate them with spaces, commas or semicolons.

2 In the **Description** edit control, type a string that identifies the file filter string. The string, which can be up to 32 characters long, will be shown in the **File Filters** list in the various file-opening dialogs.

3 You can then add the new definition to the existing list. The **Add at top** button adds the definition to the head of the list; the **Add at end** button adds it to the end; and the **Replace** button will replace the details of any filter selected in the list box at the top of the panel with the new details

You can define up to 32 filters.

Editing an existing filter

To change the details of an existing filter, take the following steps:

1 Double-click on the filter to be edited in the list box at the top of the panel, or select it and click the **Edit** button. The details will appear in the **Filter String** and **Description** edit controls at the bottom of the panel.

2 Change the filter string and/or description as you require

3 Click the **Replace** button to replace the current details of the filter with the new values. If you prefer, you can also click on the **Add at top** or **Add at end** buttons to make new entries at the head or tail of the list respectively.

Deleting an existing filter

To delete an existing filter, take the following steps:

1 Select the filter from the list box at the top of the panel

2 Click the **Delete** button to delete it

Re-Ordering Filters

The various file-opening dialogs will show the list of filters in the order that they appear in this dialog panel. If you want to change the order of the filters you have defined, take the following steps:

1 Select the filter whose position you want to change in the list box at the top of the panel

2 Click the **Up** button to move it one place up in the list, or the **Down** button to move it one place down

Any changes you make to your filters will not be stored until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. Until that time, you can abandon any changes that you don't want to keep by clicking on the **Cancel** button.

If you do not specify any of your own filters, PFE will automatically supply a default filter called "**All files**", with the filter string of "***.***". This will cause all the files in the current directory to be shown in the various file open dialogs.   However, if you specify *any* of your own filters, this default will *not* be added, and you will need to add a suitable filter of your own if you wish to be able to select a view of all existing files.

The Options Preferences *General Options* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **General Options** option from the list box on the left.

It allows you to configure various general-purpose facets of how PFE works

- Check the **Record Find/Replace strings on exit** box if you want the strings you specify in the dialogs started by the **Edit Find** and **Edit Replace** commands to be recorded at the end of your session, and made available when you next start PFE

- Check the **Check disk files for changes by other applications** box if you want PFE to test whether another Windows application, or another network user, has altered a file that you are currently edited. PFE will check whenever you return to it from running another application, and before saving any file to disk. If PFE detects that the file on disk has been altered by another application, you will be asked if you want to load the new version

- The **Ignore mouse click moving between windows** box lets you change how PFE uses the mouse to move control between edit windows. By default, the first time you click the mouse within a edit window PFE activates the window, but does *not* move the caret to the position clicked on or action a scroll bar move. If this box is checked, the caret position is changed on the first click in the window

- Check the **Do not show scroll bars in MDI workspace** box if you want always to have the maximum amount of workspace available. By default PFE acts like a standard MDI application, and shows draws scroll bars at the edges of the workspace if any edit window moves out of the visible area. Suppressing scroll bars can give a more pleasing operational style; however, to see windows that are outside the visible part of the workspace will require you to use a command such as **Window Cascade**

- Check the **Use 3D effect in dialogs** box if you want dialogs to appear with shading to give them a 3-dimensional look. By default 3D dialogs are enabled.

    You will need to exit and restart PFE to enable a change to this setting. Note that in the current version, if you are running the Intel version 32-bit Edition under Windows 3.1x with the Win32s system, or the PowerPC version of the 32-bit Edition, this box will not appear, and dialogs will not show a 3-dimensional effect.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Browse For Command Processor Dialog

This dialog allows you to browse your disks and directories to find the command processor executable that PFE will use for the **Execute DOS Prompt** and **Execute DOS Command with Capture** commands.

Steps

1. Use the **Drives** and **Directories** lists to move around your files until you find the directory containing the command processor executable.

2. Select the name of the file from the **File Name** list on the left.

3. Click the **OK** box to close the dialog. The full path name of the file you've selected will appear in the command processor edit control of the dialog panel.

The Options Preferences *MRU List* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **MRU List** option from the list box on the left.

It allows you to configure the list of the Most-Recently-Used files that PFE maintains for you and shows you on the **File menu**.

The **List Size** area lets you configure how many files are recorded in the list, and how many of these files PFE will add as selectable items to the **File** menu.

- The **Maximum number of files recorded** edit control lets you specify how many files PFE should record at the end of an edit session. You can set the value to any number between 0 and 64

- The **Number shown on menu** edit control lets you specify how many files from this list are shown at the end of the **File** menu. You can set this value to a number between 0 and 32.

The **Re-use Window and File Positioning** area lets you control how PFE uses the record it keeps of where windows showing files on the MRU list are on the screen.

- If you click the **Never** button, PFE will not use the recorded information when you open files by selecting their names from the **File** menu. The edit windows will be positioned and sized by Windows itself, using default value that normally prevent them hiding other windows. The caret will be positioned at the start of the first line of the file

- If you click the **Set in current session only** button, PFE will use the recorded information to open the window showing the file at the same position and size as when you last edited the file. The caret will be positioned at the exact location it had in the file when you last closed it. However, selecting this button means that PFE will not re-use the window position of any file that you have used a *previous* editing session but not yet in this session.

- If you click the **Set in this and previous sessions** button, PFE will use the recorded information to open the window showing the file at the same position and size as when you last edited the file. The caret will be positioned at the exact location it had in the file when you last closed it. Selecting this button means that PFE can re-use any window position saved from a previous editing session.

- If you check the **Use from dialogs as well as File menu** box, PFE will use any recorded window position information when you use the **File Open** or **File View** commands to open a file. With the box unchecked, PFE will use the recorded window position only if you open files from the MRU list on the **File Menu**

Depending on the resolution of your screen, you may find that the **File** menu becomes too large to be displayed completely if you set the number of recently-used files to be shown there to a large value. Checking the **Show list in separate column on File menu** box will reformat the menu so that the list of files appears in a separate column.

You will need to restart PFE to action any changes to the size and appearance of the MRU file list.

Changes you make with this panel will not be recorded until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Key Mapping* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Key Mapping** option from the list box on the left.

It allows you to configure various options that relate to how PFE maps keys to functions.

The **Keymap file loaded at startup** area lets you specify where PFE should look for the default key map file when it starts.

- Click the **Use default file in initialisation directory** button if you want PFE to load its default key map file from the initialisation directory defined by the **Startup** panel of this dialog (by default, this is the Windows directory). The key map file loaded will be called **pfe.key** (16-bit Edition) or **pfe32.key** (32-bit Edition)

- Click the **Use this file** button if you wish to specify an explicit file name. You can type the file name, or browse for it with the "**...**" button. Note that any file name you type will be converted to an absolute path name

Note that the file name specified here can be overridden at startup time by using the "**/k**" command line option.

To edit the key mappings that you have established, click the **Edit Key Mappings** button. This starts the dialog that edits the settings, and is equivalent to using the **Options Key Mapping** command

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Browse For Default Key Map File Dialog

This dialog allows you to browse your disks and directories to find the key map file that you want PFE to load automatically when it starts.

Steps

1 Use the **Drives** and **Directories** lists to move around your files until you find the directory containing the key map file.

2 Select the name of the file from the **File Name** list on the left.

3 Click the **OK** box to close the dialog. The full path name of the file you've selected will appear in the file name edit control of the dialog panel.

The Options Preferences *Saving Files* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Saving Files** option from the list box on the left.

It allows you to configure various options that relate to what PFE does when files are written to disk.

- Check the **Clear undo actions after saving file** if you want PFE to discard the details of how to undo the edit actions performed on a file whenever it is saved to disk. By default, undo actions are forgotten when you write a file to disk; this can reduce the amount of memory needed for the file substantially. With the box checked, undo actions are discarded only when the stack becomes full and the oldest action must be deleted to make room for a new one, or you use the E**dit Clear Undo** command

- Check the **Allow save of unaltered files** box if you want the **File Save** command and the corresponding toolbar button to be always available, whether or not the window you are working in has been altered. By default, the command can only be used if you have made some alterations that have not been preserved on disk.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Screen Font* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Screen Font** option from the list box on the left.

It allows you to configure the font that PFE uses to display text in your edit windows. At the top of the panel, a thumbnail window shows sample text that will let you see how your chosen font will appear.

- Click the **ANSI Fixed Pitch Font** button to select the standard fixed-width ANSI font. The exact font used will depend on your video driver and Windows configuration

- Click the **System Fixed Pitch Font** button to select the standard fixed-width system font. The exact font used will depend on your video driver and Windows configuration

- Click the **OEM Fixed Pitch Font** button to select the standard fixed-width OEM font. The exact font used will depend on your video driver and Windows configuration. This font will allow you to display the IBM DOS graphic characters in your windows

- Click the **Installed Fixed Pitch Font** button to select one of the fixed pitch fonts installed on your system. PFE allows you to use True Type, ATM or other installed font. The name and other details of the font selected appears in the information box below the button. If you have not yet chosen a font, you will now need to click on the "**...**" browse button. This starts a dialog that allows you to choose from your installed fonts.

Note that although PFE allows you to select from your installed fonts, it does not allow you to pick proportionally-spaced ones. Therefore, most of the fonts you have installed will not appear in the font selection dialog.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Starting PFE* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Starting PFE** option from the list box on the left.

It allows you to configure various options that relate to how PFE starts running

The **Start main window** area controls how PFE creates its main window

- Click the **As last time** button if you want the main window to be created at the same size, and in the same position, as it was when your last session ended. This is the default setting.

- Click the **Maximized** button if you want the main window to be always created to fill the whole of your screen.

The **Auto-Startup Action** area controls what, if any, action you want PFE to perform automatically when it starts.

- Click the **None** button if you do not want any action performed automatically. This is the default state.

- Click the **New empty window** button if you want an empty edit window to be created when PFE starts. This is equivalent to your using the **File New** command

- Click the **Open existing file** button if you want to be prompted to open an existing file for editing. This is equivalent to your using the **File Open** command

The **Location of initialisation files** configures where PFE looks by default for its initialisation file and its default key map file.

- Click the **In the Windows directory** button to have PFE take the files from your Windows directory. This is the default case, and is compatible with the way most Windows applications operate

- Click the **In the same directory as the PFE executable** button to have PFE look for the files in the directory that contains the **pfe.exe** or **pfe32.exe** file that you are running

- Click the **In the startup directory** button to have PFE look for the files in the directory that is the current directory when it starts. This directory can be set by configuring the PFE icon in **Program Manager**

- Click the **In this directory** button to have PFE look in a specific directory. You can type in a directory pathname, or browse for it with the "**...**" button. Note that any pathname you type will be converted into an absolute pathname.

The rest of the dialog panel contains miscellaneous items

- Check the **Only run one instance at a time** button if you want only one copy of PFE to be running at any time. This is the default state; if you try to start PFE and it is already running, the existing copy will be activated rather than a new one started.

  This setting can be overridden at startup with the "**/m**" command line option

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons

in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Status Bar* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Status Bar** option from the list box on the left.

It allows you to configure the status bar

The **Visibility** area controls whether the status bar appears at the bottom of the screen or not

- Click the **Visible** button if you want the status bar to appear

- Click the **Hidden** button if you do not want to see the status bar. This will produce extra window space for you to see your files in, but you will lose an important source of information about what PFE is doing as you edit

If you wish to configure the status bar colours, click the **Set Colours** button. This takes you to the **System Colours** panel of this dialog, and is equivalent to selecting the **System Colours** item from the list at the left of the dialog.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any changes that you have made, click the **Cancel** button.

The Options Preferences *System Colours* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **System Colours** option from the list box on the left.

It allows you to configure the colours of the application workspace (or PFE desktop); the status bar and the toolbar

A thumbnail window at the top of the panel allows you to see the effect of your choices before thay are applied, so that you can easily find a pleasing colour combination

- Click the **Use Colours set by control panel** button if you want the colours to be those you have defined with Control Panel

- Click the **Use custom colours** button if you want to specify your own colours. The list box below this button shows you the names of the various objects you can configure; you should take the following steps

    1   Double click the name of the object whose colour you want to change in the list box, or select it and click the **Change object colour** button

    2   This starts the standard colour selection dialog which allows you to choose the colour you want

    3   Repeat the steps above for the other objects you want to change.

            Note that although the colour selection dialog allows you to choose dithered colours, PFE will not use these, but will instead use the nearest solid colour. The sample areas will show you the exact background colour you have specified at the left and right edges; the text and the background immediately behind the text will show the colours that will actually be used.

Colour changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any changes that you have made, click the **Cancel** button.

Note that in this release of PFE the background colour of the pictures shown in the tool bar buttons cannot be changed from light grey.

The Options Preferences *Templates* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Templates** option from the list box on the left.

It allows you to configure various options relevant to handling templates.

- To specify the directory that PFE will search on startup for template files to be attached automatically, type the directory name or browse for it with the "**...**" button. Note that any directory name you type will be converted to an absolute pathname.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Browse For Template Directory Dialog

This dialog allows you to browse your directories to find the directory that you wish PFE to scan for template files to be attached automatically on startup.

Steps

    1 Select the drive you want to browse from the **Drive** list at the bottom right of the dialog.

    2 Move around the directories on this drive by double-clicking on their names until the directory you want to choose is shown at the top left.

    3 Click the **OK** box to close the dialog. The full path name of the directory showing at the top left will appear in the dialog panel.

The Options Preferences *Tool Bar* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Tool Bar** option from the list box on the left.

It allows you to configure various options that relate to the toolbar

The **Toolbar button size** area lets you configure the size of the buttons that make up the toolbar:

- Click the **Auto** button to have PFE set the size automatically to match your screen resolution. This is the default state.

- Click the **Small** button to use the smallest set of buttons, normally used for 640 x 480 screens

- Click the **Medium** button to use the medium sized set of buttons, normally used for 800 x 600 screens

- Click the **Large** button to use the largest set of buttons, normally used for 1024 x 768 and larger screens

The **Visibility** area controls whether the tool bar appears on the screen or not

- Click the **Visible** button if you want the tool bar to appear

- Click the **Hidden** button if you do not want to see the tool bar

The **Position** area controls exactly *where* on the screen the tool bar will appear when it it visible

- Click the **Along top edge of main window** button to show the tool bar as a horizontal strip at the top of PFE's main window

- Click the **Along bottom edge of main window** button to show the tool bar as a horizontal strip at the bottom of PFE's main window

- Click the **Down left edge of main window** button to show the tool bar as a vertical strip at the left of PFE's main window

- Click the **Down right edge of main window** button to show the tool bar as a vertical strip at the right of PFE's main window

- Click the **Free floating window** button to show the tool bar as a small window that you can position anywhere within the PFE main window. In this form, the tool bar will appear above your edit windows.

If you wish to configure the tool bar colours, click the **Set Colours** button. This takes you to the **System Colours** panel of this dialog, and is equivalent to selecting the **System Colours** item from the list at the left of the dialog.

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The toolbar buttons will not appear with the new size selected until you have exited and restarted PFE.

The Options Preferences *User Help Files* Panel

This panel appears at the right of the dialog started by the **<u>Options Preferences</u>** command when you select the **User Help Files** option from the list box on the left.

It allows you to configure up to five items that can be added on the end of the **Help** menu, which enter Windows help files of your choice.

The dialog panel allows you to specify new menu items, edit existing ones, and change the ordering of how they appear.

Defining a new menu item

To define a new help menu item, take the following steps:

1 In the **Menu item** edit control at the bottom of the dialog, type the string you want to appear on the menu. This can be up to 32 characters long.

  To set a particular letter in the string as the *hot key*, so that you can quickly select the item from the keyboard, precede the letter with a "**&**" character. To include a "**&**" character itself in the string, you must specify it as "**&&**"

2 In the **Path** edit control, type the name of the help file to be opened when the menu item is selected, or <u>browse</u> for it with the "**...**" button. Note that any file name you type will be converted to an absolute pathname.

3 You can then add the menu item to the existing list. The **Add at top** button adds it as the first of the items added to the **Help** menu; the **Add at end** button adds it as the last; and the **Replace** button will replace the details of any menu item selected in the list box at the top of the panel with the new details

You can define up to 5 items to add to the menu.

Editing an menu item

To change the details of an existing menu item, take the following steps:

1 Double-click on the menu item to be edited in the list box at the top of the panel, or select it and click the **Edit** button. The details will appear in the **Menu item** and **Path** edit controls at the bottom of the panel.

2 Change the menu item string and/or file path name as you require

3 Click the **Replace** button to replace the current details of the menu item with the new values. If you prefer, you can also click on the **Add at top** or **Add at end** buttons to make new entries

Deleting an existing menu item

To delete an existing menu item, take the following steps:

1 Select the menu item from the list box at the top of the panel

2 Click the **Delete** button to delete it

Re-Ordering menu items

The **Help** menu will show the added menu items in the order that they appear in this dialog panel. If you want to change the order of the items you have defined, take the following steps:

1 Select the menu item whose position you want to change in the list box at the top of the panel

2 Click the **Up** button to move it one place up in the list, or the **Down** button to move it one place down

Note that if you have not specified an explicit help file to be used by the **Help Context Help** command with the **Context Help** panel of this dialog, context help will use the help file specified by the first menu item in the list

Any changes you make to help menu items will not be stored until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. Until that time, you can abandon any changes that you don't want to keep by clicking on the **Cancel** button.

The Browse For Help File Dialog

This dialog allows you to browse your disks and directories to find a Windows help file to be associated with an item that you are adding to the **Help** menu.

Steps

1 Use the **Drives** and **Directories** lists to move around your files until you find the directory containing the Windows help file.

2 Select the name of the file from the **File Name** list on the left.

3 Click the **OK** box to close the dialog. The full path name of the file you've selected will appear in the **Path** edit control of the dialog panel.

The Options Preferences *Window Control* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Window Control** option from the list box on the left.

It allows you to configure various options that relate to how PFE creates file edit windows for you.

The **Open file window maximized** area lets you control the sizing of windows when you open files or use the **File New** command.

- Check the **Never** box if you want file edit windows to appear initially in a restored state. The size and positioning will be controlled by Windows, and will suit the size of the main window and any other edit windows you have open. This is the default state.

- Check the **Always** box if you want file edit windows to be created in a maximized state always. The windows will use the maximum amount of space available within the PFE main window

- Check the **If main window is maximized** box if you want file edit windows to be created maximized if and only if the main PFE window is itself maximized

- Check the **If current window is maximized** box if you want file edit windows created maximized if you both have at least one other window also in existence, and that window is itself currently maximized

- Check the **If main and current windows are maximized** box if you want file edit windows created maximized if you both have at least one other window also in existence, and that window is currently maximized, and the PFE main window is itself maximized

The **Edit window text margin** area controls the size of the small margin that PFE leaves in edit windows at the extreme left of each line. This margin improves the general appearance of the window, and makes mouse selection action easier, by moving the first character of each line slightly away from the left border of the edit window.

- Check the **Size based on system font** box if you want the margin size to be computed based on the average character width of the system font that your video driver is using. This option, which is the default, ensures that the margin is always a reasonable size, whatever you set your screen resolution to

- Check the **Set to *n* pixels** box and type in a margin width in pixels if you wish to specify an exact width to be used. The value you can type can be in the range 0 to 20 pixels.

The rest of the dialog panel contains miscellaneous options:

- Check the **Use whole main window depth for vertical tiling** if you want the **Window Tile Vertical** command to use the whole area of PFE's main window for the tiled file edit windows. By default, a small area at the bottom will be left clear to show file window icons.

- Check the **Minimize when last file window closes** box if you want the main PFE window to become an icon when you close the last file edit window that you are using. By default, the state of the main window does not change

Changes you make with this panel will not be actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Options Preferences *Windows Managers* Panel

This panel appears at the right of the dialog started by the **Options Preferences** command when you select the **Windows Managers** option from the list box on the left.

It allows you to configure the five "Managers" items that appear on the **Execute** menu and normally run the various Windows manager programs such as **File Manager** and **Print Manager**

To configure the menu items, take the following steps:

1 Select the menu item you want to configure from the **Item on Execute menu** list at the top of the panel

2 If you want the menu item to start the appropriate standard Windows manager program, click the **Run standard Windows manager** button. If you want to execute a specific command line, click the **Run this command line** button and type the command line you want executed

3 Click the **Apply change** button when you have finished configuring the menu item. The settings you have typed are noted, but are not actioned at this point

4 Repeat the steps above to configure other menu items as you require

Menu item changes you make with this panel will not be made permanent and actioned until you click either the **Apply Now** or **OK** buttons in the main dialog area to the left. If you want to abandon any alterations that you have made, click the **Cancel** button.

The Printer Font Dialog

This dialog allows you to choose the printer font that PFE will use to print text on the currently-selected printer, whose name is shown at the top of the dialog. You can choose any fixed pitch font that the printer supports.

Steps

1 Select the name of the font you want to use from the **Font** list on the left.

2 If you want some other style than plain text, such as italic or bold, select this from the **Font Style** list in the centre of the dialog.

3 Select the size of the font, in points, from the **Size** list at the right. If the font you've selected is a *scalable* font - such as a TrueType font or one generated by Adobe Type Manager - you can choose any size between 6 and 24 points. Other fonts may offer only certain sizes.

4 Select the type face effects you want from the **Effects** area:

Check the **Strikeout** box to have the text printed with a strikeout line through it

Check the **Underline** box for underlined text

5 If your printer supports colour, select the type colour from the **Color** box.

6 Click the **OK** button to select the font and close the dialog.

PFE will use the font details you've chosen whenever you print to that printer. The details are recorded and become the defaults for the printer.

The details of the printer font set up for any other printers will not be affected.

The Exit Windows Dialog

This dialog allows you to close PFE and also terminate Windows 3 at the same time.

Steps

    1 Select the way you want to close Windows

        Click the **Exit To DOS** button to close Windows and return to a DOS prompt

        Click the **Exit And Restart Windows** button to close Windows and then automatically restart it with the same options that you used previously

        Click the **Exit And Reboot System** button to close Windows and perform a warm reboot of your system

        Click the **Exit, Run DOS Command And Restart** button to close Windows, execute a single DOS program or batch file, then restart Windows. If you choose this button, enter the command line to run in the edit control below it. Note that you must give a full pathname to the **.EXE** or **.BAT** file. The **Browse** button will let you browse your disks and directories for the program to run.

        If you're running PFE for Windows/16 in the Win16 subsystem of Windows NT, all four options will simply log you off.

    2 Click the **OK** button. If any of the files you're editing have been changed, PFE will ask if you want to save the changes; you can save them, discard them, or abandon the closedown altogether.

Note that if you select the option to run a DOS command and restart Windows, and the executable is not found, or cannot be run for any reason, you will see no notification of the error.

The Exit Windows Dialog

This dialog allows you to close PFE and either log off or close Windows

Steps

1 Select the Windows closedown action you want to perform

Click the **Logoff User Only** button to simply end all the applications you've started and log you off

Click the **Logoff User And Shutdown System** button to act as above, then close Windows down so that you can power off your system

Click the **Logoff User And Reboot System** button to log you off, close down Windows and perform a warm reboot of your system

Under Windows NT, you should possess the appropriate level of privilege for all but the first action.

2 If you want to force running applications that do not respond to close automatically, rather than having Windows prompt you for each one, check the **Force Other Apps To Close** box.

3 Click the **OK** button. If any of the files you're editing have been changed, PFE will ask if you want to save the changes; you can save them, discard them, or abandon the closedown altogether.

The Browse Dialog

This dialog allows you to browse your disks and directories to find a file name that can be used by its parent dialog.

Steps

1 Use the **Drives** and **Directories** lists to move around your files until you find the directory containing the file you're interested in.

2 Select the name of the file from the **File Name** list on the left.

3 Click the **OK** box to close the dialog. The full path name of the file you've selected will appear in the appropriate edit control of the parent dialog.

The Select Window Dialog

This dialog allows you to select between many open edit windows. The list at the top of the dialog shows the window captions of some or all of the windows.

Steps

1 From the **Windows Showing** area, select the sort of window you want to see in the list:

Check the **Named Files** box to see windows showing files that have <u>associated</u> file names

Check the **Unnamed Files** box to see windows showing files that do not have associated file names

Check the **Templates** box to see windows showing <u>templates</u>

Check the **Command Output** box to see windows showing the <u>output</u> from DOS commands

2 If you want to see only windows containing changes that have not yet been saved, check the **Changed Windows Only** box

3 Either double-click the left mouse button on the name of the window you want to activate, or select it and press the **OK** button

The settings of the check boxes are remembered, and become the default for the next time you use the dialog.

The Most Recently Used Files Dialog

This dialog allows you to select a file from the list of those you've used most recently, and open it.

Steps

1   If you want to change the order in which the file names are presented in the list at the top of the dialog, click the appropriate button in the **Sort Order** area:

Click the **Last-used First** button to order the list so that the file you opened most recently is at the top

Click the **Alphabetical** button to sort the list in alphabetical order

2   If you want to see the full path name of every file in the list, check the **Show Full Pathnames in list** box. If this box is un-checked, PFE will shorten filenames as far as possible, making them relative to the current directory.

3   If you want to open the file in read only mode, check the **Read only** box

4   Normally, PFE will attempt to create the window showing the file that you open at the same size and location as it had when you last worked with it; the caret will also be positioned in the file at the place it was when the file was last closed. If you want to prevent this happening, so that the window is opend at the defaut position and size, with the caret at the left of the first line, check the **Do not use saved position** box.

4   To open a file from the list, either double-click its name in the list with the left mouse button, or select the name and press the **Open selected file** button.

5   If the file you want to open turns out not to be in the list, you can click the **Open other file** button to start the dialog that lets you select from all your files. This is equivalent to closing the current dialog and using the **File Open** command

The settings of the check boxes and sort buttons are remembered, and become the default for the next time you use the dialog.

The Colour Selection Dialog

This dialog, which is a standard part of Windows, is used by PFE in circumstances where you can choose a colour. The dialog allows you to select from a standard set of colours; to select an arbitrary colour from a colour palette; or to set up your own collection of commonly-used colours

Note that PFE does not support the use of <u>dithered colours</u>. You may select dithered colours in this dialog, but PFE will automatically convert them into the nearest solid colour.

To choose a predefined colour

To choose from a colour that has already been defined, take the following steps:

1 Click on the colour you want to select, from either the **Basic colors** panel at the top left, or the **Custom Colors** panel below it. A highlight box will appear round the selected colour

2 Click the **OK** button

To define a custom colour

If the colour you want to choose is not in the set of basic colours, you can define an arbitrary colour of any shade you wish. Note that, with most displays, all the available solid colours will be present in the basic colours set.

Take the following steps:

1 Click the **Define custom colours** button. The dialog expands, and shows you a colour selection area to the right, in which you can select the colour you want.

2 Click on one of the colour boxes in the **Custom colors** area at the lower left of the dialog. The colour you select will be shown here for later selection

3 Choose the colour you want from the area on the right. You can select the colour from the palette by moving the cross-hair cursor inside the colour square and moving the intensity slider at far right up and down; or you can type in explicit values in the **Red**, **Green**, **Blue**, **Hue**, **Saturation** and **Luminosity** edit controls.

4 The colour value you are selecting appears in the small rectangle at the bottom left of the main colour square. The left side shows you the actual colour; the right hand side shows the nearest matching solid colour, which is the value PFE will use. Normally the exact colour value will be chosen, but you can force selection of the solid equivalent by double-clicking within the area that shows it

5 Click the **Add to custom colors** button to set the chosen colour value in the **Custom colors** panel on the left. The colour is now available for you to choose as described in the first set of steps above

The custom colours you define are recorded, and will be available for selection whenever you start the dialog again.

Save Changes To File?

You have changed this file since the last time you saved it to disk. Before PFE closes it, you have a chance to update the disk copy.

Yes   Click this button to save the contents of the window to disk. If the file hasn't yet got a name, you'll see a dialog asking you to specify the file to save to.

No   Click this button to irretrievably discard the changes you've made since the last save.

Cancel   Click this button to abandon the operation completely and return to editing the window.

Save Changes To Template?

You have changed this template since the last time you stored it in the in-memory copy of a template file. Before PFE closes it, you have a chance to update the in-memory template file.

Yes   Click this button to save the contents of the window to an in-memory copy of a template file. If the template hasn't yet got a name, you'll see a dialog asking you to specify what to call it, and what template file you want to store it in.

No   Click this button to irretrievably discard the changes you've made since the last save.

Cancel   Click this button to abandon the operation completely and return to editing the window.

Storing a template in an in-memory copy of a template file does not update the template file on disk; you need to use the **Template Save File** command for this.

Cannot Detach Template File

You cannot detach a template file if you are editing one or more of the templates it contains.

Steps

1   For each of the windows containing such templates, use the **Template Store** or **Template Store As** commands to save changes you want to keep, then close the window with the **Window Close** command.

2   Preserve the changes to the template file itself with the **Template Save File** command

You will then be able to detach the template file.

Invalid Line Number

You have entered an invalid line number in the **Line To Go To** edit control. Check that what you have typed is either a numeric string, or is the word **end**.

If you have typed a number preceded by either **+** or **-** to specify a line number relative to the line that the caret is now in, check that the resulting line number is not less than 1, or greater than the number of the last line in the file

Invalid ASCII Code

You have entered an invalid ASCII code number in the dialog. Make sure that what you type is a decimal number, in the range 1 to 255.

Default Printer Not Configured

Whenever you exit from PFE, the name of the printer you last used is recorded in its <u>initialisation file</u>. This device is used as PFE's default printer the next time you start the program.

While starting this session, PFE has detected that the printer/port combination you used last time is no longer configured in your system. PFE will now use the Windows default printer unless you specify otherwise. You can check that this is what you want, and correct it if not, by using the **<u>File Print Setup</u>** command.

Replaying A Keyboard Macro While Recording

You have attempted to replay the keyboard macro that you are currently recording, which is not permitted.

You can choose to switch off the keyboard macro recorder now by clicking the **No** button. If you click the **Yes** button PFE will continue to record your keystrokes. This failed action will not be stored in the keyboard macro, and will not be repeated when you replay it.

Create A Further Backup Level

PFE has detected that the file you are about to overwrite in this operation is already in a directory that contains backups of overwritten files. You may not wish to take a further backup of the file before it is overwritten.

If you click the **Yes** button, PFE will continue as normal and take a backup copy of the file before saving the new data to it. This will involve creating a new sub-directory (called by default "**$PFEBK**") to contain it; depending on how deep your directory structure is at this point, the full path name of the directory, or of the backed up file within it, may exceed the limits imposed by the operating system.

If you click the **No** button, PFE will *not* take a backup copy of the existing file, but will simply overwrite it with the new data. The current contents of the disk file will be irrevocably lost.

If you click the **Cancel** button, the save operation will be abandoned and the disk file will not be changed.

Taking Backups Of Saved Files

Cannot Perform Command Line Substitution

The command line that you have asked PFE to execute contains substitution characters, which normally would be replaced by parts of the name of the file showing in the current window.

PFE cannot perform the requested substitutions, because either you have no files open, or because the current window does not contain a file with an associated name. You cannot substitute parts of the names of templates; and command output windows and those created by the **File New** command do not have filenames associated with them.

Command Line Is Too Long

While substituting parts of the name of the file showing in the current window into the command line to be executed, the resulting command became too long for the 512-byte buffer that PFE uses to store it.

You should change the command so that it does not exceed this size. Note that although PFE allows up to 512 bytes for the string, this may exceed the size allowed by the operating system or command processor that will run the command.

Invalid Substitution Point Encountered

While substituting parts of the name of the file showing in the current window into the command line to be executed, PFE encountered an unrecognised substitution symbol and could not continue.

Check the command line you have specified and correct the invalid substitution request. All substitution points refer to parts of the full path name of the file showing in the current window, and are these:

%d	The directory part of the path name, without a trailing "\\"
%e	The extension part of the path name, without a leading "."
%f	The filename and extension parts of the path name, separated by a "."
%n	The filename part of the path name
%p	The entire path name
%u	The drive part of the path name, followed by a "**:**"
%%	A single "**%**" character

Clear Undo Actions

This command will clear the details of all the previous editing actions that you have recorded for the current file.

If you click the **NO** button, PFE will abandon the action and no change will be made

If you click the **YES** button, PFE will clear the details it has recorded. This will result in some memory being freed for other uses (depending on what editing actions you have performed); but you will not be able to undo any changes you have made up to this point.

Context Help Not Available

PFE is unable to give context help on the item selected in the current window, because you have not specified the help file that is to be searched.

You can specify the help file with the **<u>Options Preferences</u>** command.

The Status Bar

This area of the screen is the **status bar**, which provides you with useful information on the state of PFE itself and on the current file.

For a full description of the status bar, select the **About the status bar** topic at the end of this item.

To see interactive help on any of the boxes within the status bar, use the **Help Screen/Menu Help** command to turn on interactive help, and click the left mouse button once with the box of interest.

About the Status Bar

Status Bar - File Position Area

This area of the status bar reports the position of the caret in the current file, in terms of the line number within the file, and the column within the line.

Double-clicking the left mouse button in this area will execute an **Edit Goto Line** command, which starts a dialog that lets you move quickly to an arbitrary line in the file.

If you don't have any files open, this area shows you the PFE's version number.

Status Bar - Lines In File Area

This area of the <u>status bar</u> tells you how many lines there are within the current file. If you don't have any files open, it will be blank.

When a file is being loaded, the number in this area will steadily increase, to show you how many lines have been processed so far. When you save a file, the number will decrease to zero, showing you how many lines are still to be written to disk

Status Bar - File Change Marker

This area of the status bar tells you whether the current file has been changed. If it has, the area will display a '**#**' character; if not, or if you don't have any files open, the area will be blank.

Status Bar - File Read-Only Marker

This area of the <u>status bar</u> tells you whether the current file has been made *read-only*, and so cannot be altered.

If the file is read-only, the area will show the characters "**RO**". If you are able to change the file, it will show "**RW**".

Double-clicking the left mouse button in this area will change the file's state from *read-only* to *writable* and vice-versa.

Status Bar - Prefix Key Area

This area of the status bar tells you whether you have started a two-key command sequence by typing one of the configured prefix keys.

If you have, the area shows you the mnemonic of the prefix key you've used. If you haven't started a two-key command, or you have no files open, the area will be blank.

Status Bar - Macro Recording Area

This area of the status bar tells you whether keystrokes are currently being recorded in a keyboard macro.

If they are, the area will contain the text "**Rec on**". If you're not recording keystrokes, it will contain the text "**Rec off**". The area will be blank if you don't have any files open.

Double clicking the left mouse button in this area will turn recording on or off.

Status Bar - Text Wrap Area

This area of the status bar tells you whether text wrapping is active in the current window.

If wrapping is not turned on, the area will show the text "**No wrap**". Otherwise, it will show the word "**Wrap**", followed by the number of the window column at which wrapping will occur.

The area will be blank if you don't have any files open.

Double clicking the left mouse button in this area will turn wrapping on or off.

Status Bar - Save Format Area

This area of the <u>status bar</u> tells you the format in which the current file will be saved if you write it to disk.

If the area shows the text "**DOS**", the file will be written with each line terminated by a **Carriage Return** byte and a **Line Feed** byte (**CR-LF**), which is the standard for MS-DOS files.

If the area shows the text "**Unix**", the file will be written in Unix format, with each line ended by a single **Line Feed** byte (**LF**).

The area will be blank if you don't have a file open.

Double clicking the left mouse button in this area will change the format.

Status Bar - Typing Mode Area

This area of the status bar tells you the typing mode of the current window.

If the area shows the text **"INS"**, characters you type will be inserted, displacing any characters already in the file to the right. If it shows the text **"OVR"**, characters you type will overwrite those already there.

Double clicking the left mouse button in this area will change between the two modes.

Status Bar - Num Lock Area

This area of the <u>status bar</u> tells you the current state of the **Num Lock** key. It will show the text **"NUM"** if Num lock is active.

Status Bar - Caps Lock Area

This area of the <u>status bar</u> tells you the current state of the **Caps Lock** key. It will show the text **"CAP"** if caps lock is active.

Tool Bar Button

This tool bar button lets you quickly perform actions such as creating a new file edit window. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Ctrl** or **Shift** keys down at the same time.

Left click alone

> This executes a **File New** command to create a new, empty edit window

Left click + Shift

> This executes a **Template New** command to create a new, empty template window

> **About the Tool Bar**

Tool Bar Button

This tool bar button lets you quickly perform actions such as opening existing files. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Ctrl** or **Shift** keys down at the same time.

Left click alone

This executes a **File Open** command to open an existing file for editing

Left click + Shift

This executes a **File View** command to open an existing file in *read-only* mode

Left click + Ctrl

This executes a **Template Edit** command to edit an existing template

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly perform actions such as writing files to disk. To use it, position the mouse cursor on the button and click the left button once.

This executes a **File Save** command to write the current file to disk. If the current file is a template rather than a file, a **Template Store** command is executed to save it to a template file.

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly cut the selected text in the current window to the clipboard. To use it, position the mouse cursor on the button and click the left button once.

This executes an **Edit Cut** command to delete the highlighted text from the current file and transfer it to the clipboard.

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly copy the selected text in the current window to the clipboard. To use it, position the mouse cursor on the button and click the left button once.

This executes an **Edit Copy** command to copy the highlighted text to the clipboard. The current file is not changed by this operation.

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly paste the contents of the clipboard into the current window. To use it, position the mouse cursor on the button and click the left button once.

This executes an **Edit Paste** command to paste the clipboard into the current window at the position of the caret.

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly perform search actions in the current file. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Shift** key down at the same time.

Left click alone

> This executes an **Edit Find** command to show you the dialog that controls searching for text

Left click + Shift

> This repeats exactly the last action you performed with the **Edit Find** command

> **About the Tool Bar**

Tool Bar Button

This tool bar button lets you quickly perform text replacement actions. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Shift** key down at the same time.

Left click alone

This executes an   that starts the dialog controlling text replacement actions

Left click + Shift

This repeats exactly the last text replacement operation you performed with the **Edit Replace** command

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly undo the last edit action. To use it, position the mouse cursor on the button and click the left button once.

This executes an **Edit Undo** command to reverse the effects of the last edit action you performed.

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly insert a <u>template</u> into the current window. To use it, position the mouse cursor on the button and click the left button once.

This executes a **<u>Template Insert</u>** command, which will show you a dialog from which you can select the template to insert.

<u>About the Tool Bar</u>

Tool Bar Button

This tool bar button lets you quickly perform actions related to <u>keyboard macros</u>. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Shift** key down at the same time.

Left click alone

This executes a **<u>Macro Start Recorder</u>** command to start recording if not already started; or a **<u>Macro Stop Recorder</u>** command to stop it.

Left click + Shift

This executes a **<u>Macro Replay</u>** command to replay the last-recorded keyboard macro.

<u>About the Tool Bar</u>

Tool Bar Button

 This tool bar button lets you quickly start a DOS command processor session. To use it, position the mouse cursor on the button and click the left button once.

This executes an **Execute DOS Prompt** command to start a command shell

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly initiate DOS commands such as compilers, capturing their output in a window. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Shift** key down at the same time.

Left click alone

> This executes an **Execute DOS Command To Window** command, which shows you a dialog allowing you to specify the command line to be executed.

Left click + Shift

> This executes an **Execute Repeat DOS Command To Window** command to repeat the last command executed with **Execute DOS Command To Window**.

> **About the Tool Bar**

Tool Bar Button

This tool bar button lets you quickly launch Windows applications, or configure the list of <u>Windows tools</u>. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Shift** or **Ctrl** keys down at the same time.

Left click alone

This executes an **<u>Execute Launch Application</u>** command, which starts a dialog allowing to specify the command line for the application to be run.

Left click + Shift

This executes an **<u>Execute Launch Windows Tool</u>** command to start a dialog allowing you to launch a pre-configured Windows tool.

Left click + Ctrl

This executes an **<u>Execute Configure Windows Tools</u>** command to start a dialog which allows you to set up the details of the Windows tools you want to use.

<u>About the Tool Bar</u>

Tool Bar Button

This tool bar button lets you quickly turn line numbering on or off in the current window. To use it, position the mouse cursor on the button and click the left button once.

Line numbering will be turned off it is currently on, and on if it is currently off.

About the Tool Bar

Tool Bar Button

This tool bar button lets you quickly initiate printer-related actions. To use it, position the mouse cursor on the button and click the left button once, possibly holding the **Shift** key down at the same time.

Left click alone

This executes a **File Print** command which starts a dialog allowing you to print the current file.

Left click + Shift

This executes a **File Print Setup** command which starts a dialog allowing you to configure details of the printer to use.

About the Tool Bar

Recently Used File

This menu item contains the name of a file that you have opened recently. PFE maintains a <u>list</u> of the files you open, and automatically adds them to the end of the **File Menu**.

Clicking the left mouse button on one of these file name items will open the file for editing.

You can configure the size and appearance of the list with the **Options Preferences** command.

More Files/Recent Files

This menu item is part of the list that PFE maintains of files that you have opened recently. You can configure the number of files that PFE records, and if you set this number to more than PFE can display on the end of the **File Menu**, this menu item will be added.

Clicking the left mouse button on the command will start a dialog that will allow you to choose any file from the complete list of those opened recently.

Active Window

This menu item contains the title of one of the windows that are currently open. PFE automatically adds the titles to the end of the **Window Menu**.

Clicking the left mouse button on one of these window titles will make the corresponding window active.

If you have more than 9 windows open, an additional **More Windows** item will appear on the end of the menu; selecting this item shows you a dialog that lets you select from all the titles.

More Windows

This menu item is added to the end of the **Window Menu** whenever you have more than 9 windows open simultaneously.

Selecting this item shows you a <u>dialog</u> that lets you select from the titles of all the open windows and make one of them the active window.

Exit Windows

This system menu item allows you to terminate PFE, and to shut down your Windows system in a variety of ways.

The command starts a <u>dialog</u> that allows you to specify various closedown options

Save Screen

This system menu item allows you to start a Windows screen saver immediately, instead of having to wait for the configured inactivity period.

This option will not work with screen savers that do not use the standard Windows screen-saver interfaces to function.

The System Menu

The system menu contains a number of standard commands that allow you to manipulate PFE's main window and change to other applications. The items at the start of the list are common to all Windows applications.

Close

Terminates your PFE session

Move

Allows you to move PFE's main window with the keyboard cursor keys

Size

Allows you to resize PFE's main window with the keyboard cursor keys

Minimize

Makes PFE's main window into an icon

Maximize

Expands PFE's main window so it fills the entire screen

Restore

If PFE's main window is an icon, restores it to its previous windowed state

Switch To

Starts the Windows Task Manager to allow you to shift to another application

File Position Specifications

The **Edit Goto Line** command, the **EditGotoLine()**   DDE command, and the **-g** command line option, allows you to specify a location within a file to move the caret to. This specification can be a simple line number, or it can include a character position *within* the line.

From the command dialog you can specify either an absolute position, or one relative to where the caret is at the moment; the command line **-g** option accepts only an absolute position

In its simplest form, a file position specification is simply a number, such as "**518**". You can specify the first line of a file as either "**1**" or "**start**", and can specify the last line of the file either by its actual line number, or by the word "**end**"

To specify a relative line number position, precede the number with either "**+**" or "**-**". Thus "**+38**" means "38 lines down from where the caret now is", and "**-7**" means "7 lines up from where the caret now is".

If you specify only a line number, PFE will always position the caret at the far left of the target line. However, you can if you want specify a character position the line also, so you can move to any arbitrary position.

Character positions are specified following the line number, and are separated from the by a "**/**" character. Like line numbers, they can be absolute values or relative ones, and you can use the words "**start**" and "**end**" for the far left and far right ends of the line respectively.

Normally, character positions are specified as *window columns* on the screen. Thus, a specification of

        58/32

would indicate line 58, in window column 32. The left-most window column is numbered 1.

If the window column specified is within the spaces formed by expanding a TAB character, PFE adjusts the position to the first character following. If the column is past the end of the line, the specification is invalid and the command will fail.

In many cases there is no ambiguity in this position; however, if your file contains TAB characters, a specific character on a line may not be in a predictable window column - the TAB expansion width may change.

To allow you to specify a position clearly whan TABs are involved, PFE also allows you to specify the position within a line by giving the character *count* within the line. This value is independent of TAB expansion, since a TAB character counts 1, the same as any other.

To indicate that you are specify a position by character count, you must precede the number with "**C**" or "**c**". Thus the specification

        37/C94

indicates line 37, 94 characters from the start of the text, counting each TAB character as 1. The position may or not be in the 94th *window* column.

The first character in the text of a line is numbered 1.

The line number component of a position specification cannot be omitted. Therefore, if you need to specify a movement *within* the *current* line, you need to specify the line number in relative form. For example, you would specify window column 27 in the current line as

```
        +0/27
```

```
58
```

Indicates line 58, at the start of the line

```
+35
```

Indicates 35 lines down from the line containing the caret now, at the start of the line

```
27/6
```

Indicates line 27, in window column 6 (or as close as TAB expansion permits)

```
+0/end
```

Indicates the end of the line now containing the caret

```
38/C45
```

Indicates line 38, at the 45th character of text, where a TAB character always counts as exactly 1 character regardless of TAB expansion width. The result may or may not be the same as that obtained by the specification `38/45`

```
127/+5
```

Indicates line 127, 5 window columns to the right of where the caret is in its present line

```
end/end
```

Indicates the far right of the last line in the file

A word is defined as a sequence of alphanumeric characters, delimited at each end by white space or punctuation. If the window has been set with a window mode of "C language" the underscore character is treated as alphanumeric

To represent characters that you can't type in the **Edit Find** and **Edit Replace** dialogs, and in the search string specified in the **EditFind()** DDE command, use this notation:

> **\f** represents a Form Feed character
> **\t** represents a TAB character
> **\n** represents the end of a line
> **** represents a single '\' character

To represent an arbitrary character code, use the notation "**\xnn**", where "**nn**" represents two hexadecimal digits. The null value "**\x00**" is not permitted.

PFE can capture the output of any DOS program that writes to the standard output or standard error files **stdout** and **stderr**. Output displayed by writing directly to video memory can't be captured.

MAPI is the electronic mail protocol devised by Microsoft and used in their Microsoft Mail application. Other vendors such as Novell use different mail systems; currently PFE does not support them

Depending on the video display adapter that is fitted to your computer, only a few of the infinite number of possible colours that exist can be shown exactly, and all the rest must be *synthesised* from combinations of others. In practice, you would probably have 16 or fewer *solid* colours available.

Colours that cannot be shown exactly are represented by a process called *dithering*, whereby Windows will create the appearance of a colour by generating a pattern of two or more other colours. For example, a pale blue could be synthesised from a grid of dark blue lines on a white background; from a suitable distance it the effect is of the colour required.

Dithered colours look extremely unattractive when used as backgrounds for text, and so PFE will convert any dithered colour you choose into the closest matching solid colour.

PFE records the details of your configuration options (as set by the **Options Preferences** command), and various items that need to be carried over between sessions, in its *initialisation file*.

The initialisation file is normally held in your system's Windows directory. The 16-bit Edition uses a file called **pfe.ini**, and the 32-bit Edition uses a file called **pfe32.ini**. You can change the location of the directory containing the initialisation file with the **Options Preferences** command.

You should not normally alter the initialisation file yourself. However if you do need to do so, you should make sure that PFE is not running, and use the standard **notepad** program to perform the edit.

To establish a DDE link to the 16-bit Edition of PFE use the service name **PFE**. To link to the 32-bit Edition, use the service name **PFE32**

This help file was created for Programmer's File Editor version 0.06.001. Some items within it may not apply to earlier or later releases.