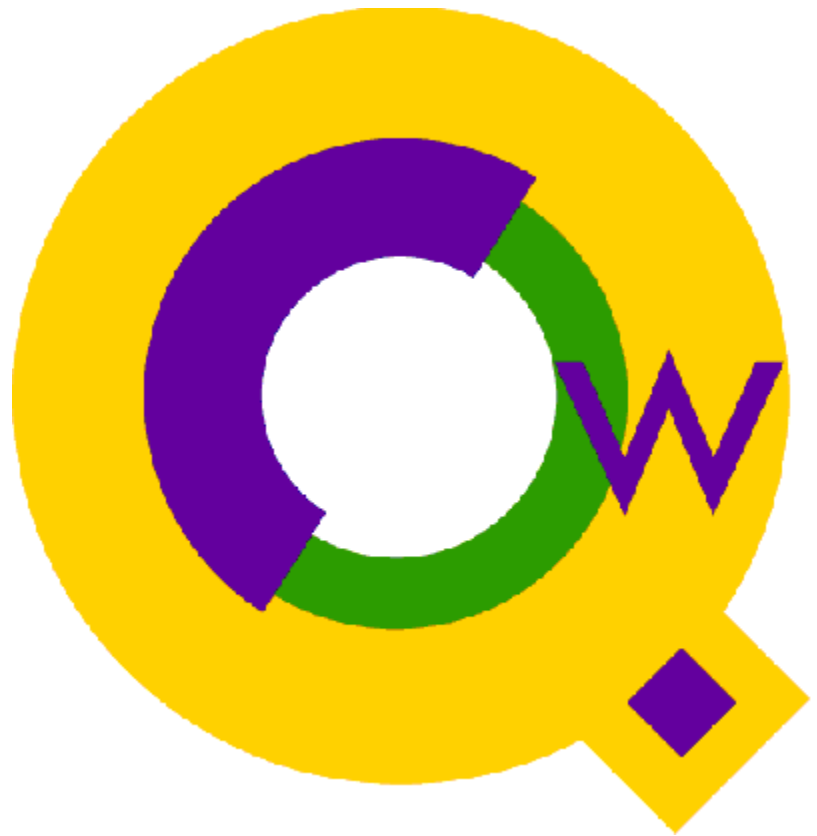

Qedit for Windows 5.9.13

User Manual

by Robelle Solutions Technology Inc.



Copyright 1996-2009 Robelle Solutions Technology Inc.

Qedit is a trademark of Robelle Solutions Technology Inc. Windows is a trademark of Microsoft Corporation. Other product and company names mentioned herein may be the trademarks of their respective owners.

Updated Saturday, April 07, 2012

Robelle Solutions Technology Inc.
7360 – 137 Street, Suite 372
Surrey, BC, Canada V3W 1A3
Toll-free: 1.888.robelle
Tel: 604.501.2001
Fax: 604.501.2003
support@robelle.com
www.robelle.com

Contents

Welcome to Qedit for Windows	1
Overview of Qedit for Windows	1
Hardware and Software Requirements	1
Network Requirements	2
The Ultimate Editing Tool	2
Multi-File Access—Simultaneously	2
Speed, Speed, and More Speed	2
Highly Scaleable	2
Fully Integrated with Windows	3
Multiple Independent Views	3
Consistent Interface on All Platforms	3
Internet Friendly	3
Host-Based Qedit vs. Qedit for Windows	3
Getting Help	4
What’s New in This Version	4
Highlights in Version 5.9.13	4
Highlights in Version 5.9.12	4
Highlights in Version 5.9.11	4
Highlights in Version 5.9.10	4
Compatibility	4
Known Problems	5
Document Window Displayed Incorrectly	5
Getting Started	7
Installing Qedit for Windows	7
Downloading Qedit for Windows	7
Installing Qedit on Stand-Alone PCs	7
Installing Qedit on Networked PCs	8
Installing the Qedit for Windows Server	9
Running the HP-UX Server	10
Support Files	11
Manuals	11
Language Templates	11
Upgrading Qedit for Windows	12
Uninstalling Qedit for Windows	12
Customizing the Client	12
The Preferences Dialog Box	13
Customizing the Server	19
How To Specify To the Server Program	19
Configuration Files on the Host	20
Connection Template File	21
Running Qedit	21
Expiring Version	21

Tip of the Day.....	22
Startup Process	22
The Application Window	23
Application Icon and Control Menu.....	23
Application Title Bar	24
Application Menu Bar	24
Application Toolbar.....	25
Toolbar Buttons.....	27
Application Workspace	30
Application Status Bar.....	30
The Document Window.....	31
Document Title Bar	32
Document Workspace	33
Document Status Bar.....	33
Splitting a Document into Independent Views.....	35
Collapsing Multiple Views into One.....	36

A Windows Primer 37

Windows Standard Functions.....	37
Switching Between Applications.....	37
Controlling the Size and Position of Windows.....	37
Moving a Window Around on the Screen.....	37
Resizing a Window.....	37
Minimizing a Window.....	38
Maximizing a Window.....	38
Arranging Windows into Tiles	38
Closing a Window	38
Mouse Operations.....	38
Using Mouse Buttons	39
Selecting Text with a Mouse	40
Selecting More Text.....	41
Extending a Selection.....	41
Rectangular Selections	42
Selecting a Menu Command.....	43
Selecting Toolbar Buttons	43
Selecting from a List.....	43
Shortcut Menus in Windows 95	43
Shortcut Menu on Text Boxes.....	43
Shortcut Menus on Toolbar Buttons.....	44
Shortcut Menu on Document Workspace.....	45
Send To Function in Windows 95	46
Adding Qedit as a Send To Target	46
Registering File Types in Windows 95.....	47
Registering Text Files.....	47
Registering Qedit Scripting Language Files.....	48
Drag-and-drop File Open.....	48
Drop on Desktop Icon	48
Drop on Qedit Application Window	49
Drop on Taskbar Button.....	49
Special and International Characters	49
Long File Names.....	49

Connecting to a Host or Network 51

What do you need to create a connection?	51
Connecting Through a Network	51
Accessing a File Server	52
Creating a Connection	52
Creating a Temporary Connection	52
Creating a Saved Connection	52
Copying a Saved Connection	56
Opening a Connection	56
Opening a Temporary Connection	56
Opening a Saved Connection	57
Editing a Saved Connection.....	58
Modifying a Saved Connection.....	58
Renaming a Saved Connection.....	58
Deleting a Saved Connection	58
Closing a Connection.....	59
Customizing a Connection.....	59
Automatically Logging In at Startup.....	59
Closing Connections without Open Files	59

Creating & Opening Files 61

Creating Files.....	61
Creating a New Host File	61
Creating a New Local File.....	62
Defining File Options	63
Setting the Line Termination Character	66
Notation in Host Operating Systems	67
UNIX Notation.....	67
MPE Notation.....	68
POSIX Notation on MPE	68
Host Directories.....	68
Accessing Host Directories	69
The Directory Listing	69
Sorting the Directory Listing.....	70
Searching for a File	70
Using Wildcards.....	70
Changing Directories.....	71
Selecting Files	71
Opening Selected Files.....	72
Printing a Directory Listing.....	72
Saving Host Files	72
Opening a File	72
Document Attributes Saved by Qedit.....	72
Opening a File from a Command Line.....	73
Opening Embedded Files	74
Opening MRU (Most Recently Used) Files	74
Opening a Local File	75
Opening an HTML File.....	77
Opening a Host File.....	77
Opening Files with Binary Data.....	80
Server Warnings.....	80
Maintaining and Using a List of Favorites	81
Viewing the List of Open Files.....	85
Customizing Opening a File	85
Setting Default Options for New Files	85

Loading Local Files into Memory	86
---------------------------------------	----

Typing, Selecting & Editing Text **87**

Overview.....	87
Determining Your Position in a File	87
Cursor Position and Appearance	87
Controlling the Insertion Point Movements	88
Insert Mode vs. Overwrite Mode	88
Typing Text and Inserting Special Characters.....	89
Typing Text.....	89
Inserting Control Characters.....	89
Inserting Nonprinting Characters	90
Nonprinting Characters for Reflection Users	91
Inserting Special and International Characters	92
Selecting Text	92
Cursor Position and Appearance During Selection	93
Multiple Ways to Select Text	93
Copying, Moving and Pasting Text	94
The Clipboard.....	94
Copying Text.....	96
Pasting Text.....	96
Moving Text.....	96
Deleting Text	98
Merging Two Lines	98
Working with Columns and Tabs	98
Deleting Text in Columns.....	99
Inserting Text in Columns	99
Shifting Text in Columns	100
Shifting to the Next Tab Stop.....	101
Removing or Inserting Tab Characters.....	101
Expanding Tabs in Clipboard.....	102
Working with Rectangular Selections	102
Copying and Cutting a Rectangular Selection.....	102
Pasting a Rectangular Selection in Replacement.....	102
Inserting a Rectangular Selection	103
Deleting or Clearing a Rectangular Selection	103
Extending a Rectangular Selection.....	103
Undoing and Redoing	103
What Delimits an Undo and Redo Transaction?	104
Undo Command.....	105
Redo Command.....	105

Text-Formatting Features **107**

Overview.....	107
Fonts	107
Changing the Document Font.....	107
Changing the Directory Font.....	108
Bold, Italics and Underline	108
Ruler Bar.....	109
Displaying and Hiding the Ruler Bar	110
Tab Stops	110
Tab Stops on the Server.....	110
Tab Stops on the Client	111

Auto-Indent.....	114
Guide Lines.....	114
Setting Guide Lines on the Ruler	114
Moving or Clearing Guide Lines on the Ruler	115
Setting Guide Lines in the Properties Dialog Box	115
Moving Guide Lines in the Properties Dialog Box	117
Clearing Guide Lines in the Properties Dialog Box	117
Invisible Formatting Characters.....	117
Matching Opening and Closing Characters	118
Working With Syntax Coloring.....	119
Ambiguous Language Identification	119
SQL Language Templates	119
Order of Precedence	120
Customizing Text-Formatting Features	120
Displaying the Ruler Bar When Opening Files	120
Setting the Default Font for Document Windows	121
Setting the Default Value for Auto-indent	121
Displaying Invisible Formatting Characters When Opening Files	121
Setting the Default Color Scheme	121

Line Features 127

Overview	127
Record Numbers	127
Host Line Numbers	127
Record Numbers.....	127
Trailing Blanks	128
Preserve Trailing Blanks	128
COBOL Tags.....	129
COBOL Tags in Server Configuration Files	129
COBOL Tags in the Client.....	131
Line Length.....	133
Maximum Line Length.....	133
Line Termination	133

Searching & Replacing Features 135

Overview	135
Scrolling Through a File.....	135
Using the Scroll Bars.....	136
Using the Intelligent Mouse	136
Using Cursor Movement Keys	137
Setting Markers in a File.....	137
Marking a Position	137
Returning to a Marked Position.....	137
Exchanging Current Position with Last Marker	137
Using the Last Marker to Select Text.....	138
Defining the Terms of a Search	138
Common Search Operations	139
Finding Consecutive Occurrences of a String	139
Finding the First and Last Occurrence of a String	140
Finding a Highlighted Word	140
Limiting a Search to Part of a File.....	140
Searching in Columns	141
Finding a Line Number in a Host File.....	141

Finding a Record Number	142
Common Replace Operations	142
Replacing Occurrences of a String	142
Replacing All Occurrences of a String	143
Replacing a String Based on Pattern Matching	143
Replacing a String Using Regular Expressions	143
Customizing Your Search Options	143
Allowing the Caret in Undefined Areas	143
Automatically Marking Successful Searches	144
Scrolling Method for Host Files	144
Setting Default Search Options	145

Copying & Printing Files 147

Overview.....	147
Copying a File.....	147
Copying Local Files	147
Copying Host Files	148
Converting a File to Qedit Format.....	150
Convert Qedit Workfile to Text Format.....	151
Printing on a Local Printer.....	151
Print Options on a Local Printer.....	151
Printing on a Local Printer	153
Printing on a Host Printer	154
Print Options on a Host Printer	154
Changing the Default Printer in MPE.....	155
Printing on a Host Printer	156

Saving & Closing Files 157

Overview.....	157
Saving a File	157
Which files do you need to save?	157
Auto-Saving Local Files.....	157
Saving a Qedit Workfile.....	157
Saving an Existing File (Local and Host).....	158
Saving a New Host File	158
Saving a New Local File	159
Saving All Opened Files.....	160
Closing a File.....	160
Explicit vs. Implicit Close	160
Closing a File.....	160
Forcing a Close on a Host File	161
Customizing Saving and Closing Files.....	161
Automatically Posting Qedit Workfiles on the Server	161
Comparing Timestamps Before Overwriting Host Files	162
Saving Changes at Regular Intervals.....	162

Interacting with Hosts 163

Overview.....	163
Executing Host Commands.....	163
Single-Line Execution	163
Multi-Line Execution	166
Disabling Host Command Execution	167
MPE Commands Execution.....	167

User Defined Commands (UDC)	167
Session Environment	167
Process-dependent Settings	168
Session Initiation and Termination	168
Duplicate Output Lines	168
UNIX Commands Execution	169

Script-Based Features **171**

Overview	171
Script Libraries	172
Managing Scripts	173
Monitoring and Controlling Scripts	173
Robelle Script Library	174
Sorting Lines	175
Listing Lines	176
Compiling MPE Source Files	178

Power Tips for Developers **181**

Overview	181
Spanning Hosts and Operating Systems	181
Switching and Tiling Files	181
Switching Between Files	181
Tiling Document Windows	182
Working Smarter with a Terminal Window	184
Advantages of Qedit Workfiles	184
Automatic Save	184
Sole Access to Files	184
Preventing Accidental Overwrite of Files	184
Instant Access to Embedded Files	185
Getting the Fully Qualified Name of a File	185
Handling Nonprinting Characters	185
Minimizing Keystrokes	185
Menu Shortcuts	185
Letter Matching	185
Expanding the Workspace	186
Quick Access to Directories	186
Working with MPE Hosts	186
Working with Copylibs	186
User Labels on Files	186

Network Issues **189**

TCP/IP	189
Firewalls	189
Client to Server Handshaking	189
Looking at Server Connections	191
MPE Connection List	191
UNIX Connection List	192

Reference **193**

Menu Commands	193
File Menu	193
Edit Menu	195

Text Menu	196
Mark Menu	197
View Menu	197
Script Menu	197
Options Menu	198
Window Menu	199
Help Menu	199
Key Combinations and Functions	200
Control Combination Keys	200
Searching Shortcuts	200
Editing Shortcuts	201
Other Shortcuts	201
Function Keys	202
Troubleshooting	203
Overview	203
What to do in case of problems?	203
Getting a Trace	204
Server Trace Files	206
Changing Debug Settings	206
File Information Dialog Box	206
Document Information	207
Connection Information	208
Transmission Information	209
Server Settings	209
Cache Statistics	210
Connection Problems	211
MPE Host Connections	211
Firewall Protocol on MPE Connections	213
HP-UX Host Connections	216
Connection Progress Messages	218
Recovering a File	219
Recovering Temporary Host Files	219
Local Files Operation	221
Time Zones	221
Document Database Recovery	225
Error Messages	226
Appendix A - Earlier Highlights	229
Overview	229
Appendix B - File Types	231
File Codes and Mnemonics	231
Appendix C – Regular Expressions	243
What Are Regular Expressions?	243
Metacharacters	243
Anchor Characters	244
Match Any Character	244
Optional Character	245
Repeating Characters	245
Character Class	245

Negated Character Class	246
Repeating Character Class.....	246
Escape Character	246
Escaped Sequences in Regular Expressions	247
Backreferences in Regular Expressions.....	247

Appendix D - Language Templates **251**

Overview	251
Template Location	251
Which Language?	252
Language Template Format.....	252
Language Block.....	253
File Extensions	254
String Delimiters	254
Continuation Schemes	255
Comment Schemes	255
Token Character Set	256
Keywords	257

Glossary of Terms **259**

Index **265**

Welcome to Qedit for Windows

Overview of Qedit for Windows

Welcome to Qedit for Windows, the Microsoft Windows editor that edits local and server files, both large and small. Use Qedit for Windows to edit files on your local PC, network computers that are available from Microsoft Windows, and files on your HP-UX or MPE servers. Throughout this manual, the terms HP e3000 and MPE will be used interchangeably to identify an **MPE** host. Likewise, the terms UNIX, HP 9000 and HP-UX will be used to identify a **UNIX** host.

Hardware and Software Requirements

Qedit for Windows up to version 4.7.01 is available in a 16-bit edition for Windows 3.1 and 3.11. As a pure 16-bit version, it does not require any of the files from the Win32s subsystem. *Versions 4.7.02 and later are only available in the 32-bit edition.*

Qedit is also available in a 32-bit edition for Windows 95 and Windows NT. Qedit for Windows requires a minimum of 4 MB of memory and approximately 3 MB of disk space.

If you want to use Qedit to access corporate servers from home or remote offices, you might have to reconfigure your firewall software or router. The connection to the HP-UX server occurs on port 7395. On HP e3000 servers, port 1542 is the default Remote Process Management (RPM) listener. You can also configure individual MPE connections to use Qedit's firewall protocol. If you use this option on MPE, the MPE server will work like an HP-UX server and Qedit will use port 7395 as the default.

If you use an Internet Service Provider (ISP), you might be assigned a random IP address when you connect from home. You have to make sure you can connect using all possible IP addresses.

The Qedit server software has to be installed on your HP-UX or MPE servers. This software communicates with Qedit for Windows running on your local PC to provide fast service for all your editing needs.

To get accurate time information in the **Directory** dialog box, you have to make sure the time zones on all your computers are set correctly. See "Time Zones" on page 221.

Network Requirements

To edit HP-UX or MPE server files, you must have a TCP/IP connection configured in Microsoft Windows. Qedit requires a Winsock 1.1 compliant TCP/IP stack. If you use a Web browser (e.g., Netscape Navigator or Microsoft Internet Explorer), you already have TCP/IP installed and configured.

Qedit works best on a true TCP/IP connection. Qedit should work with proxy servers. This is often used in corporate intranets to give people access to the Internet without exposing their actual IP address. When using proxy servers, you will likely have to use Qedit's **Firewall protocol** on MPE connections.

The Ultimate Editing Tool

Qedit for Windows is your ultimate editing tool. Now you can get the editing functions you need in one easy-to-use package. Qedit's efficient use of resources lets you edit both local and host files with lightning speed, even over a network or the Internet. You can seamlessly span multiple platforms concurrently, and use keystroke shortcuts and toolbar buttons to quickly maneuver through your files. In the familiar environment of Microsoft Windows, Qedit's ultimate advantage is that by using it to its full potential it helps you reach yours.

Multi-File Access—Simultaneously

Qedit for Windows allows you to have multiple MPE, HP-UX, and local files open at the same time. You can work on all these platforms, seamlessly cutting and pasting information between any number of files.

Qedit for Windows can access several files at once. In fact, the maximum number is only limited by the amount of Windows resources you have available. These can be a mix of local files and remote files, new files and existing files, text files, data files and Qedit workfiles. Each host connection is also capable of opening several files on that host.

Speed, Speed, and More Speed

Do you work over a slow modem? Is your local area network clogged with Web traffic? Don't worry about Qedit for Windows—it makes minimal demands on your network. When editing a host file, searches are actually done on the server, not the client. You can quickly search through enormous files because Qedit does not download the entire file to your PC, only the portions in which a match is found. Speed is one of Qedit's main advantages. Faster access and faster processing means that you can accomplish more every day.

Highly Scalable

No matter what grows—your file size, the number of users, the traffic on a network, or the geographical area of users—Qedit's performance does not deteriorate. It has the flexibility to adjust to any situation. With Qedit, the tasks you do today do not limit your future capabilities. Whether your procedures are simple or complex, you can count on Qedit's ability to adapt to your needs.

Fully Integrated with Windows

Qedit for Windows is a true Windows product that lets you copy and paste information to and from other Windows applications. Open a host file, select a portion of it, and paste it into Microsoft Word. Start Word, select some text, and paste it into a Qedit for Windows host file. All of this is possible with Qedit for Windows.

Multiple Independent Views

How many times have you wished you could look at your variable declarations while working on some new code in the middle of the program? How about working on a file with very wide lines where you can never see the left and right margins together on the screen? Qedit's split view feature can give you the whole picture by splitting a document window vertically or horizontally. All the panes are created with their own set of scroll bars and work independently from each other.

Consistent Interface on All Platforms

Qedit is capable of editing both local PC files and files on your host computers such as the HP e3000 and the HP 9000. If you are used to the full-screen interface of Qedit for HP-UX and Qedit for HP e3000, you might expect the same interface in Qedit for Windows. That is *not* what we decided to do. We chose to make the user interface of Qedit for Windows look as much as possible like the standard Windows editors such as Microsoft Notepad, Microsoft Wordpad and Microsoft Word. This makes the interface the same for all files you edit and reduces training costs. (Perhaps with the exception of those whose fingers have the Qedit for HP e3000 or HP-UX keystrokes burned into them.)

As much as possible, a local file looks and acts like an MPE file, which looks and acts like a UNIX file. There are, however, a couple of areas where the flavor of the host-based Qedit slips into Qedit for Windows.

Internet Friendly

The client/server design works equally well on Internet connections. People working at home can connect their PCs to the Internet through a public Internet Service Provider and from the Internet to the company LANs. However, it might be necessary to reprogram a firewall to allow the Qedit for Windows connections from the external Internet to the internal LAN. The design of the Qedit for Windows protocol makes fast response editing possible, even over very slow network connections.

Host-Based Qedit vs. Qedit for Windows

There are features of regular (host-based) Qedit which are not yet available in Qedit for Windows. Although we are implementing the most popular features as fast as possible, the dedicated Qedit host user will still occasionally need to switch to a terminal window and run regular Qedit.

Getting Help

This user manual is also available in Windows Help format. You can display Help information by using the **Contents** command of the **Help** menu or by pressing F1. All dialog boxes include a Help button, which can display context-sensitive information (i.e., it takes you directly to the section that discusses the current dialog box).

The **Help** menu also provides access to Robelle resources available over the Internet. Use one of the commands in the **Robelle Web** submenu to connect your browser to Robelle's Web site (<http://www.robelle.com>) or see the latest features in Qedit for Windows. You can also use the **E-mail Technical Support** command to send a question or comment to our Technical Support staff.

What's New in This Version

The current version of Qedit for Windows is 5.9.13. Here are a few of the changes and enhancements made since version 5.9.

Highlights in Version 5.9.13

- Under certain circumstances, an Undo or Redo action could cause an abort.

Highlights in Version 5.9.12

- The `newconnectiontemplate` QSL method created incorrect MPE connection. The session password was incorrectly used as the user password.

Highlights in Version 5.9.11

- It is now possible to specify the number of files appearing in the various filelists: **Recent Files**, in the **Favorites** list and individual **Favorites** subfolders. This is done in the **General** tab of the **Preferences** dialog.

Highlights in Version 5.9.10

- Qedit for Windows does not abort anymore when editing a Cobx file on a Unix server. The abort occurred when a number of conditions were present.

For a history of development highlights, see "Appendix A - Earlier Highlights" on page 229. For Qedit Scripting Language (QSL) highlights, refer to the *QSL User Manual*.

Compatibility

In most cases, when you have selected some text, typing a character replaces the selection. With the introduction of column operations, there is now one exception to this rule. If you have a selection and you hit the TAB key, the selected lines are shifted right. The text is not replaced. If you want to replace the selection, you have to use the DELETE key (or any other key except TAB) first to get rid of the selection then enter the tab character.

Known Problems

Here is a list of problems that were known at the time the version was released. These problems will be fixed in future versions.

Document Window Displayed Incorrectly

You might run into a display problem when minimizing document windows to an icon. If you work on other documents and restore the icon to its previous size, the window might appear without borders, taking only part of the application workspace. This is a minor problem but does not affect editing operations. To correct this problem, simply maximize the window.

Getting Started

Installing Qedit for Windows

The installation process for Qedit for Windows is the same as in any other Windows application. Qedit is available as a download from Robelle's website. There are two basic types of installations: **Typical** for stand-alone PCs and **Network/Workstation** for network file servers.

If you have virus-checking software, it might prevent the installation process from copying Qedit program files to your PC or file server. If this is the case, you might have to temporarily disable the virus-checking software.

Downloading Qedit for Windows

Versions of Qedit for Windows can be downloaded from Robelle's website. The website also provides installation instructions.

There are 2 types of files: production and pre-release versions. There is a production version every year. Pre-releases are intermediate versions created between production versions. In all cases, the download file is called qinstall.exe which is a self-extracting executable.

Installing Qedit on Stand-Alone PCs

For each stand-alone PC, you need to install Qedit as a **Typical** installation. With stand-alone PCs, you have to re-install new versions of Qedit on each PC that has a licensed copy. If you have a large number of PCs on a network and all of them are connected to a file server, the **Network** installation method is much easier to upgrade.

To perform a **Typical** installation on a stand-alone PC,

1. Start the installation program, qinstall.exe.
2. Follow the instructions on the screen.
3. When you are asked for the type of installation, select **Typical**.
4. When asked, specify the destination drive and directory on which you want to install Qedit. You can use the default location provided by the installation program.

5. Next, you are prompted for the name of the Program Folder where the Qedit shortcuts should be created. You can use the default location or enter your own.
6. To validate your copy of Qedit, enter the **Serial Number Verification** codes received from Robelle.
7. To start Qedit right after it has been installed, simply select **I would like to launch Qedit for Windows** in the last step of the installation.

Installing Qedit on Networked PCs

On PCs connected to a file server, you need to perform a two-part installation process. First you have to perform a **Network** installation on the file server. Then on each PC connected to the network, you have to perform a **Workstation** installation.

To upgrade to a new version of Qedit, however, you only need to repeat the **Network** installation. PCs connected to the network will automatically run the new version. Although this two-part process is a little more time-consuming when Qedit is first installed, it saves much more time with subsequent upgrades.

Network Installation

To perform a **Network** installation on the file server,

1. Log on to the file server either directly or through a PC with proper access.
2. Copy the self-extracting executable to the file server.
3. Start the installation program, qinstall.exe.
4. Follow the instructions on the screen.
5. When you are asked for the type of installation, select **Network**.
6. Next, specify the destination drive and directory on which you want to install Qedit on the **Network** server. Make sure this directory is on a path accessible to other Qedit users.
7. The installation program asks for the name of a Program Folder where the Qedit shortcuts should be created. You can use the default location or enter your own.
8. When the setup program asks you to validate your copy of Qedit, enter the **Serial Number Verification** codes received from Robelle.

Workstation Installation

Make sure you are connected to the file server on which you performed the **Network** installation. To perform a **Workstation** installation on each PC,

1. If you have copied the installation files on the **Network** server,
 - Click the **Start** button on the taskbar and select **Run**.
 - Type in the full program path. For example,
`<drive>:\robelle\qedit\qinstall.exe`
2. Follow the instructions on the screen.
3. When the procedure asks for the type of installation, select **Workstation**. A Workstation installation does not copy any files to the local PC. It simply adds the necessary program group and icons for the Qedit program on the server.

- When asked, specify the directory in which the **Network** version of Qedit has been installed on the server. You do not have to enter the **Serial Number Verification** codes.
- To start Qedit right after it has been installed, simply select **I would like to launch Qedit for Windows** in the last step of the installation.

Installing the Qedit for Windows Server

The Qedit for Windows server is also available for download from Robelle's website.

If you are planning to install the HP-UX server, you have to make sure the Qedit daemon is not running. To determine if it is or not, you can do the following:

- Log in as root on the system console or on a terminal
- Enter

```
ps -ef | grep qedit
```

This command looks at all the processes currently running on your system and selects only the ones that contain the string “qedit.” You should get a listing similar to this:

```
root 165 240 2 12:05:29 ttyp4 0:00 grep qedit
joe 140 122 0 09:16:57 ? 0:10 /opt/robelle/bin/qedit -d -t
root 122 1 0 Oct 15 ? 0:00 /opt/robelle/bin/qedit -d -t
cybil 544 122 0 06:44:22 ? 0:01 /opt/robelle/bin/qedit -d -t
ricky 159 122 0 Oct 10 ? 0:01 /opt/robelle/bin/qedit -d -t
```

The last column on the right shows the name of the programs selected. The line that reads “grep qedit” is the command you used to get this listing. Ignore it. All the other ones (running /opt/robelle/bin/qedit -d -t) are Qedit server instances.

- The line that shows a one (1) in the third column is the Qedit daemon process. All the other processes are Qedit for Windows user connections. In this example, these connections are logged in as joe, cybil and ricky. If you know who these users are, you should contact them and ask that they disconnect until the installation is complete.
- Repeat the command in step 2 until the Qedit daemon is the only process left running. This is the process you want to ultimately remove. To do so, you have to use the HP-UX `kill` command. In this example, you would enter:

```
kill 122
```

If you are planning to install the MPE server, you have to make sure the server program is not used by anyone either as a host-based editor or as a server with **Firewall** protocol. To determine if the program is accessed, on newer versions of MPE, you can do:

- `listf qedit.pub.robelle,8`

```
*****
FILE: QEDIT.PUB.ROBELLE
6 Accessors(O:6,P:6,L:0,W:0,R:6),Share
#S2506 RICKY,CLERK.ACCTG P:1,L:0,W:1,R:1 LDEV: 15
#S2504 GL,CYBIL.DEV P:1,L:0,W:1,R:1 REM :
24.201.21.203
#S2503 AR,JOE.DEV P:1,L:0,W:1,R:1 REM :
207.42.133.226
#S2501 BOB,OPERATOR.OPER P:1,L:0,W:1,R:1 LDEV: 20
```

```
#J8351 QSERVER,MGR.ROBELLE P:1,L:0,W:1,R:1 SPID:
#O2552
```

2. The lines with "LDEV" are typically sessions running host-based Qedit. If you know who these users are, you should contact them and ask they exit from Qedit until the installation is complete.
3. The lines with "REM : " are typically remote users connected over the network. They are likely to be Qedit for Windows users. You should contact them and ask they disconnect until the installation is complete.
4. The line starting with "#J" is a job running on the host. It could be a job using Qedit to perform some file edits or a Qedit server running with **Firewall** protocol. There is not easy way to tell which one it is. If it's a not a Qedit listener, simply wait for the job to complete. If it is a Qedit listener, simply abort the job. In this example, you would use:

```
abortjob #j8351
```

Running the HP-UX Server

1. If you are installing the server on a HP-UX host, refer to the Qedit for HP-UX User Manual for a complete set of installation instructions or perform the following steps.
2. Once the HP-UX server files have been copied to the HP-UX host, you need to sign on to the host as `root` using a terminal or emulator (e.g. WRQ's Reflection).
3. Set the `ROBELLE` environment variable required by the Qedit server.

```
ROBELLE=/opt/robelle
export ROBELLE
```

4. Start the Qedit for Windows daemon

```
/opt/robelle/bin/qedit -d -t
```

The server process keeps running until it is stopped explicitly with the `kill` command or until the system is shut down. You have to do steps 2 to 3 every time you want to start the server. If you want to start the daemon as part of the system startup process, you can get detailed instructions in the Qedit for HP-UX User Manual or you can contact Robelle technical support.

The `-t` option enables the server trace facility. If the Qedit daemon is started with this option, all user connections will produce trace messages also. If the trace facility is not enabled on the daemon, the trace facility for individual users can be enabled by checking the **Enable trace** option of the **Server Trace Options** on the **Debug Controls** dialog box.

5. If you need to install the server programs on multiple hosts, simply repeat these steps for each one.

Support Files

Up to version 5.8.10, support files were created in the system root directory (usually \Windows on Windows 95 and XP or \Winnt on Windows NT)

If you install Qedit in its default location, files required by Qedit are copied to a directory called \Robelle\Qedit on the drive you specified during the installation. These files include the tips file, the system script library and language templates. Other support files are created in the Windows user's application data directory. Typically, this is \Documents and Settings\loginUsername\Application data\Qedit. The support files are:

- Qwin.ini

This file contains all global configuration settings, such as options in the **Preferences** dialog box, the debug options, the UNIX port number, the MPE server name and the default fonts.

- Qedconn.dat

This file contains information about all your connections. For security reasons, confidential information such as passwords is encrypted. This is the default name. The connection template file location can be changed in the **Configure Server Settings** dialog box.

- Qederr.log

This file contains the last error message for any internal consistency failure. This can sometimes help track down the cause of a problem when tracing is not enabled.

- Qwinddb.dat

This is the document database. It contains a number of document-specific attributes. For recovery purposes, Qedit for Windows makes a copy of the document database in QWINDDB.bkp.

- Qwintips.txt

The Qedit tips file provides helpful suggestions on how to make the best use of Qedit. This file is created in the same directory as the main program file.

Manuals

The Qedit for Windows User Manual, Qedit Scripting Language User Manual and the Quick Start Guide are available in Adobe PDF files. They are called **qwin.pdf**, **qslidoc.pdf** and **qstart.pdf** respectively. These files are in the **Manuals** subdirectory under the directory where Qedit has been installed. During installation, a shortcut to the **Manuals** subdirectory is created in the **Robelle** program group of the Windows Start menu.

You can view these files with Adobe Acrobat Reader. Acrobat Reader is available for free and can be downloaded from numerous Web sites such as <http://www.adobe.com>. You can use Acrobat Reader to print your own copies of the manuals.

Language Templates

The syntax coloring feature requires language template files. Template files are formatted according to XML specifications. They must have a ".XML" extension.

These files are stored in **Language** subdirectories in different locations. Robelle provides language templates in `\robelle\qedit\system\language`. These templates should not be modified as they might be overwritten when a new version is installed. If you wish to create your own templates or customize one of the Robelle templates, you should create them in `\robelle\qedit\user`. Typically, the `user` subdirectory contains files that have been customized and should be shared by all Qedit users. The `system` and `user` directories are always scanned.

If you have templates that are only useful to you, you can store them in the directory of your choice. Enter the location in the **User Language Directory** field on the **General** tab of the **Preferences** dialog box.

Upgrading Qedit for Windows

If you already have Qedit for Windows and are upgrading to a more recent version, all your support files are preserved. This means you do not have to create all your connections again; all your preferences and default settings are still active and document attributes are still applied.

Up to version 5.8.10, Qedit support files were installed in the Windows system root directory (`\windows` or `\winnt`). Starting with version 5.8.11, the support files are stored in a subdirectory called `Qedit` under the Windows user's application data directory. This is usually `\Documents and Settings\loginUsername\application data\Qedit`.

The directory name is based on the `APPDATA` system variable. If the variable does not exist, the support files remain in the system root directory.

For a new install, the subdirectory is created the first time Qedit is launched. For an upgrade, the install script copies the current support files to the new location, if they do not exist already. The original files are not removed but they are not used anymore. The install script copies default support files only.

- The application profile: `QWIN.ini`
- The document database and its backup: `QWINDDB.dat`, `QWINDDB.bkp`
- The connection template file: `QEDCONN.dat`. If the connection template file was changed in the **Customize Server Settings** dialog box, that file is not copied.
- The error log file: `QEDERR.log`.

Uninstalling Qedit for Windows

If you want to remove Qedit from your PC, you can use the **Add/Remove Programs** function in the **Control Panel** of Windows 95. Select Qedit for Windows from the application list.

Customizing the Client

Whether you have one user or one hundred users, Qedit for Windows can create a comfortable environment for everyone. You can customize various features in the Qedit client to make your daily tasks as efficient as possible. Each user's settings are

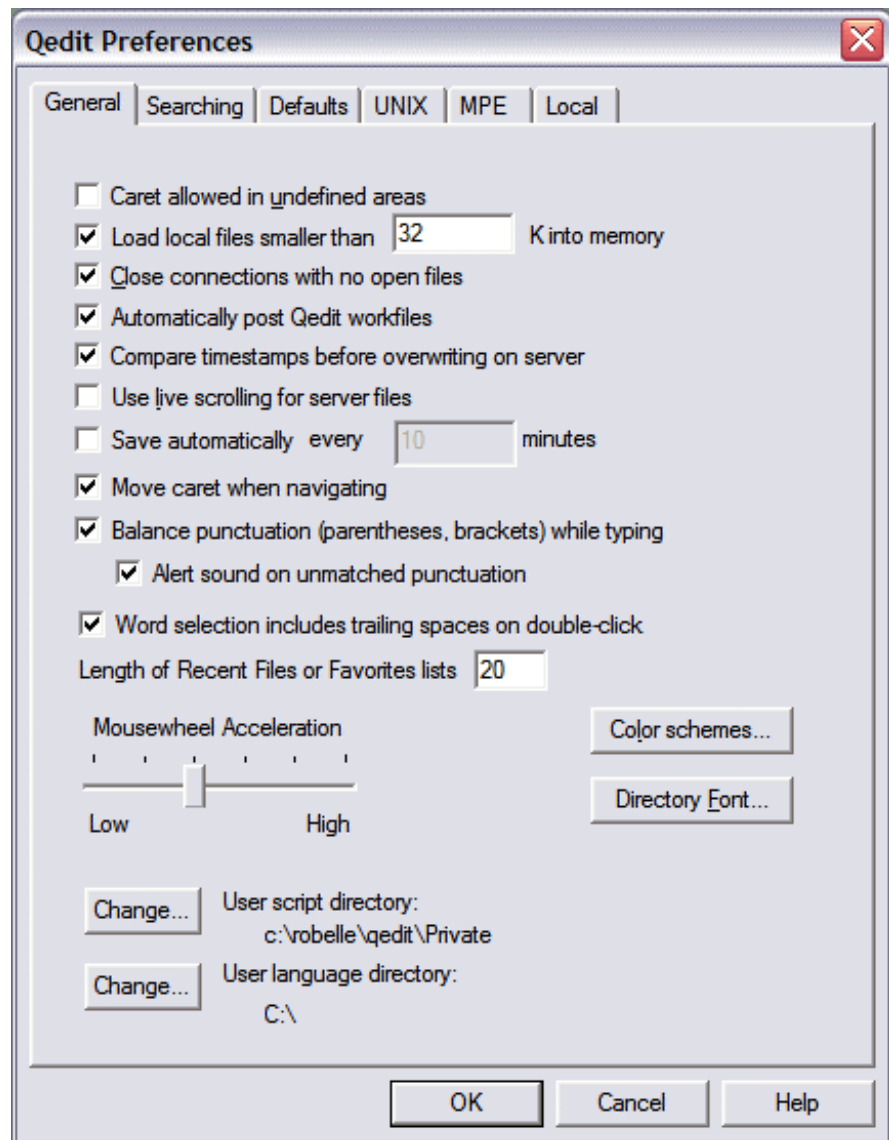
saved in the **Qedit Preferences** dialog box. Choose preferred search defaults, navigation tools, font settings and file options. These default settings are saved until you decide to change them.

The Preferences Dialog Box

Shortcut: ALT+O,P

To bring up the global **Qedit Preferences** dialog box, choose the **Preferences** command in the **Options** menu. This dialog box contains six tabs: General, Searching, Defaults, UNIX, MPE and Local. Use the TAB key to move between options on the same tab. To move from one tab to another, simply click the tab name.

The General Tab

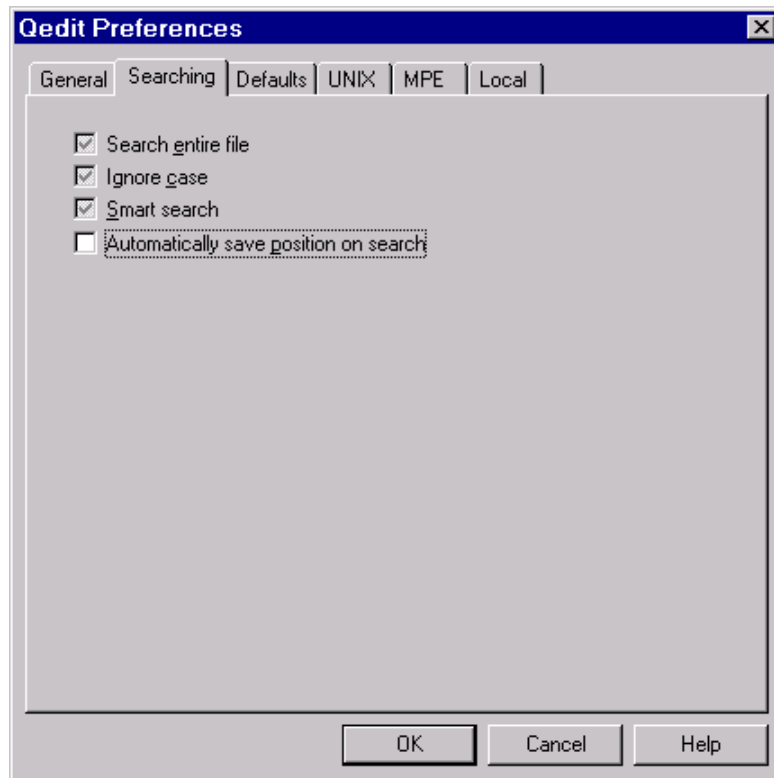


General settings in the Qedit Preferences dialog box

For a description of the options on the **General** tab, go to the following sections:

- **Automatically post Qedit workfiles, Compare timestamps before overwriting on server and Save automatically**, see "Customizing Saving and Closing Files" on page 161.
- **Balance punctuation (parentheses, brackets) while typing and Alert sound on unmatched punctuation**, see "Matching " on page 118.
- **Caret allowed in undefined areas**, see "Allowing the Caret in Undefined Areas" on page 143.
- **Close connections with no open files**, see "Customizing a Connection" on page 59.
- **Color schemes**, see "Customizing Text-Formatting Features" on page 120.
- To change the **Directory font**, see "Changing the Directory Font" on page 108.
- **Length of Recent Files or Favorites lists**, see "Opening MRU (Most Recently Used) Files" on page 74 and "Maintaining and Using a List of Favorites" on page 81.
- **Load local files smaller than ... K into memory**, see "Customizing Opening a File" on page 85.
- **Mousewheel Acceleration** slider, see "Using the Intelligent Mouse" on page 136.
- **Move caret when navigating**, see "Controlling the Insertion Point Movements" on page 88.
- **Use live scrolling for server files**, see "Scrolling Method for Host Files" on page 144.
- **User Script Directory**, see "Script Libraries" on page 172.
- **User Language Directory**, see "Language Templates" on page 11.
- **Word selection includes trailing spaces on double-click**, see "Selecting Text with a Mouse" on page 40.

The Searching Tab

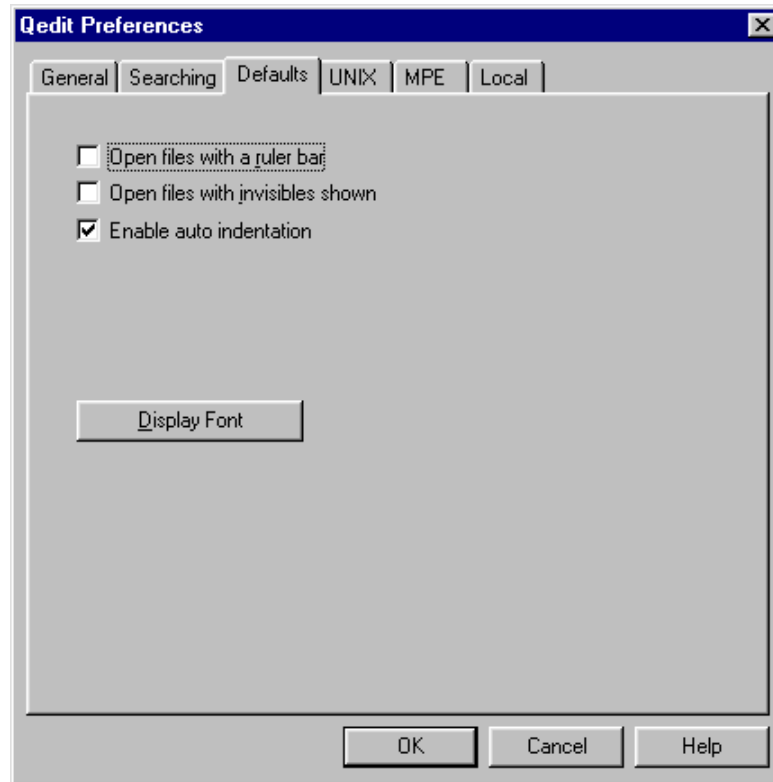


Search settings in the Qedit Preferences dialog box

For a description of the options on the **Searching** tab, go to the following sections:

- To set **Searching** preferences **Search entire file**, **Ignore case** and **Smart search**, see "Customizing Your Search Options" on page 143.
- To set **Automatically save position on search**, see "Customizing Your Search Options" on page 143.

The Defaults Tab

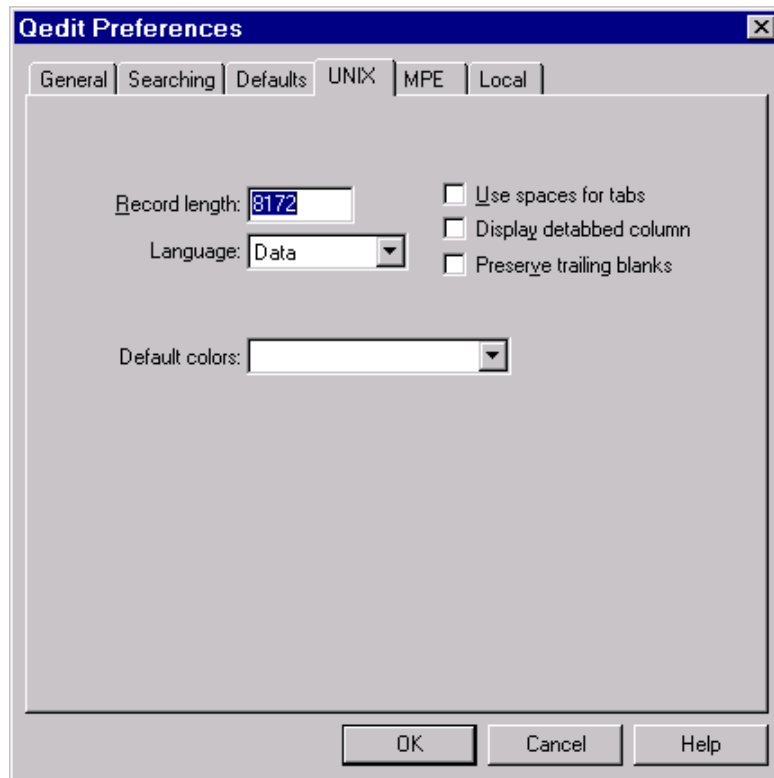


Defaults settings in the Qedit Preferences dialog box

For a description of the options on the **Defaults** tab, go to the following sections:

- **Open files with invisibles shown**, see “Customizing Text-Formatting Features” on page 120.
- **Open files with a ruler bar**, see “Customizing Text-Formatting Features” on page 120.
- **Enable auto indentation**, see "Customizing Text-Formatting Features" on page 120.
- To set the **Display font**, see “Customizing Text-Formatting Features” on page 120.

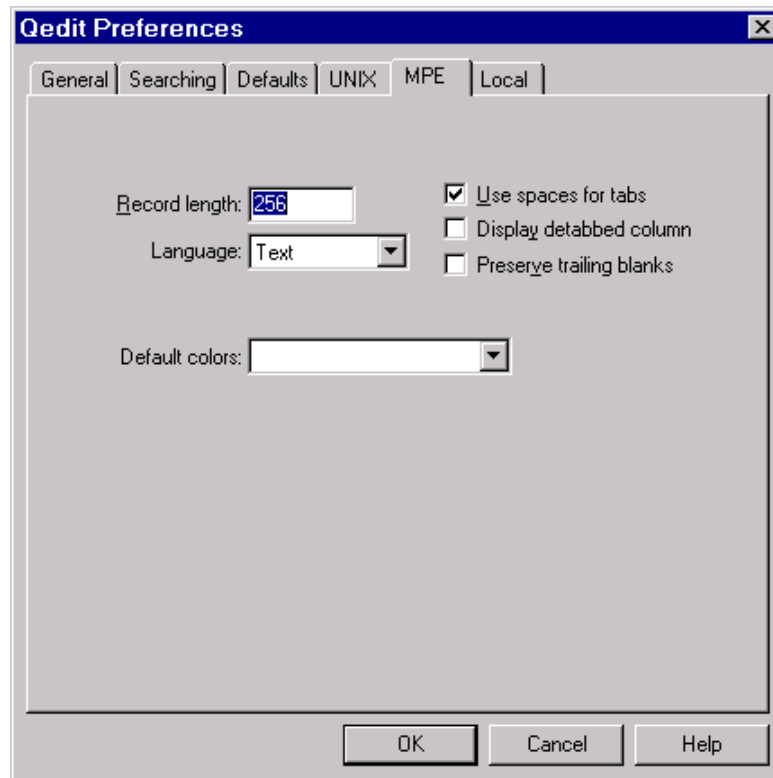
The UNIX Tab



Default UNIX file characteristics in the Qedit Preferences dialog box

- For a description of the options on the **UNIX** tab, go to "Setting Default Options for New Files" on page 85.

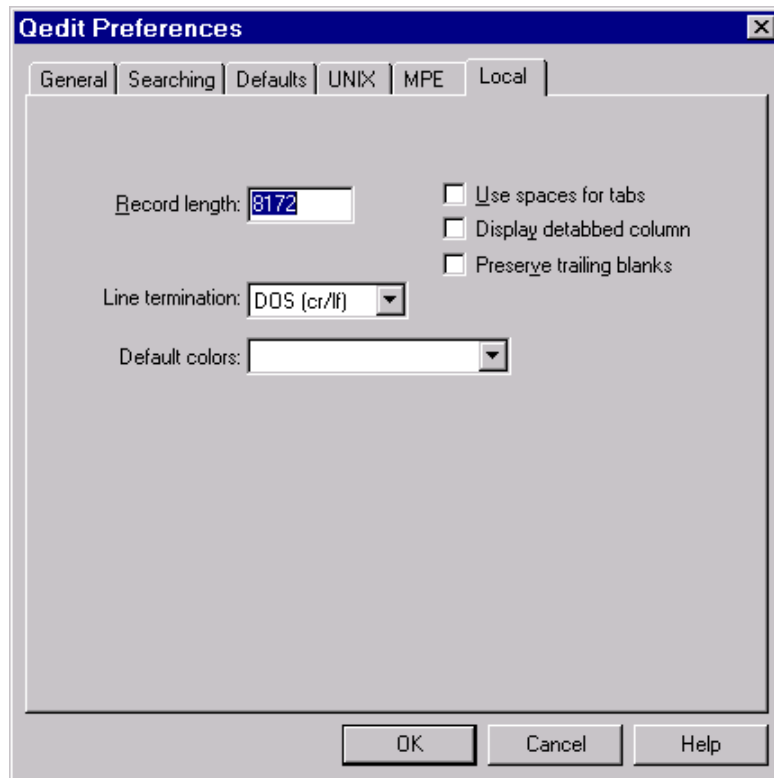
The MPE Tab



Default MPE file characteristics in the Qedit Preferences dialog box

- For a description of the options on the **MPE** tab, go to "Setting Default Options for New Files" on page 85.

The Local Tab



Default local file characteristics in the Qedit Preferences dialog box

- For a description of the options on the **Local** tab, go to "Setting Default Options for New Files" on page 85.

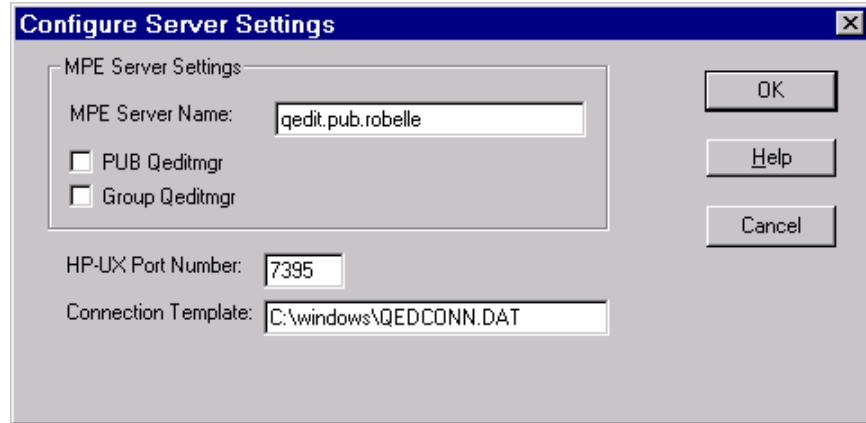
Customizing the Server

When working with files located on host computers, Qedit for Windows uses a client-server architecture. This means that the client program running under Windows interacts with a server program running on the host computers. Qedit has to know how to contact the server and the server's behavior can be customized directly on the host.

How To Specify To the Server Program

Qedit needs to know how to connect to the various servers. For normal connections to MPE hosts, you have to specify the actual name of the server. By default, Qedit tries to execute `Qedit.Pub.Robelle`. This is default value for **MPE Server Name** text box.

For MPE connections using the **Firewall protocol** and UNIX hosts, you have to specify the port number on which the server is listening to. The default value for **HP-UX Port Number** is 7395.



Configure Server Settings dialog box

Configuration Files on the Host

You can also customize the behavior of the server by putting appropriate commands in a configuration file on the host.

The Qedit server always executes the commands in system configuration files such as `Qeditmgr.Pub.Robelle` on HP e3000 and `/opt/robelle/qeditmgr` on HP-UX. Commands in these configuration files are executed for anyone using Qedit on the host machine either as the host-based editor or as the server. Refer to the appropriate host-based *Qedit User Manual* for details.

On MPE hosts, other Qeditmgr configuration files can be installed in the `Pub` group of the logon account and in the logon group for a particular connection. Execution of these files is optional and can be enabled in the **Configure Server Settings** dialog box.

You can enable one or both options to use the corresponding configuration file. If they are both enabled, the server executes the files in the sequence shown (Pub first then the logon group) immediately after the system configuration files. The Pub and group Qeditmgr files are typically used to further customize a user's environment.

On UNIX hosts, users can also have personal configuration files called `.qeditmgr`. The server looks for this file in the user's home directory. Unlike MPE, these personal configuration files are *always* executed; there is no option to enable.

Using the Qedservmode Variable

The QEDSERVMODE variable on the hosts allows you to have different settings for the server version and for the host-based version of Qedit. QEDSERVMODE is always set to zero if you are in the regular mode and to non-zero if you are in server mode. If you are using the Qedit server as well as regular Qedit on the same host, you should modify the Qeditmgr file so that commands written for a particular mode are only executed in that mode. This is particularly true on MPE hosts because logon User Defined Commands (UDCs) and command files are not executed in sessions initiated by Qedit for Windows. This might cause some commands in Qeditmgr to fail (e.g., non-existent variables). Ultimately, the server environment might not be set correctly.

For example, on the HP e3000 you could perform


```
if Qedservmode = 0 then
  set extprog mpex.pub.vesoft
endif
```

On UNIX, you would check the value of QEDSERVMODE in a script executed from the Qeditmgr file, because Qedit/UX does not support the If/Then command.

Connection Template File

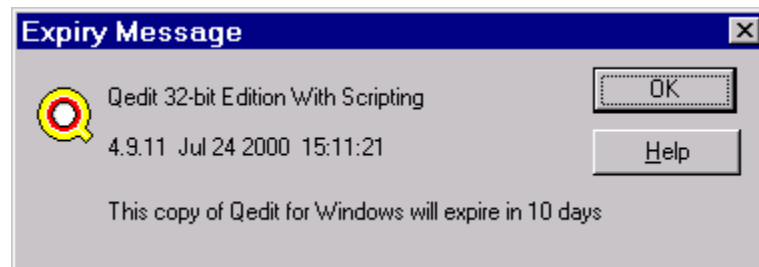
All the host connections that you create are stored in a file on your local hard-disk. The default is to store them in a file called `Qedconn.dat` in the Windows directory. You can change the connection template file name in the **Connection Template** text box.

Running Qedit

Once Qedit for Windows has been installed, you can start it by clicking the appropriate icon in the **Robelle** program group.

Expiring Version

If you have an evaluation copy of Qedit and the version is about to expire, you might see the following **Expiry Message** dialog box.

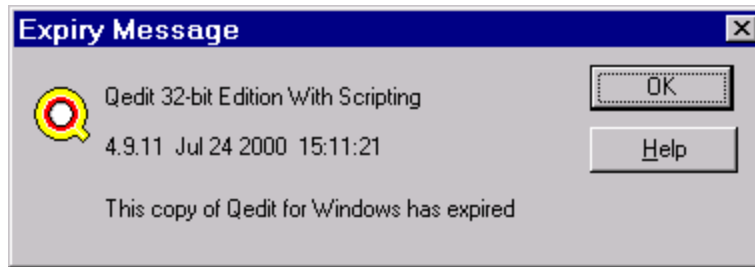


Expiry Message dialog box

You can see the details about the version you are running:

- 32-bit edition (Windows 95 and NT)
- the version number and build timestamp
- the number of days before it expires

At the end of the trial period, you will not be able to use Qedit at all. You will get the following message indicating that the program has expired. You can call our office if you think your version should not be expiring.



Expiry Message dialog box

Tip of the Day

Qedit offers a number of useful tips. These tips are stored in a file called Qwintips.txt. By default, Qedit displays a new tip every time you start the program.



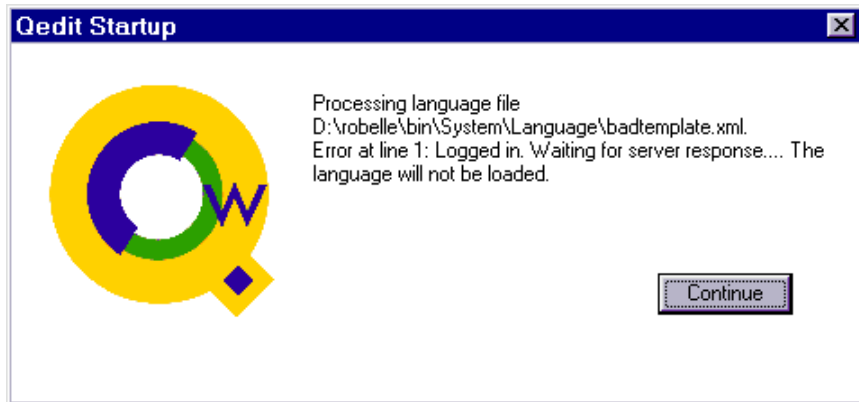
Tip of the Day dialog box

Select **Close** to return to Qedit and proceed with editing files. If you want to continue to browse through the tips, select **Next Tip**. If you do not want to see a new tip every time you start Qedit, disable the **Show Tips on StartUp** option.

You can also browse through the tips by using the **Tip of the Day** command in the **Help** menu. This command displays the same dialog box.

Startup Process

Upon startup, Qedit searches for QSL scripts to load automatically. It also loads language templates used for syntax coloring. During this operation, progress messages are displayed in the Qedit Startup dialog box. If there are no errors, the dialog box is discarded. If errors are detected, an error message is displayed along with the name of the problem file. You have to click **Continue** to acknowledge the error and have Qedit continue with the startup process.

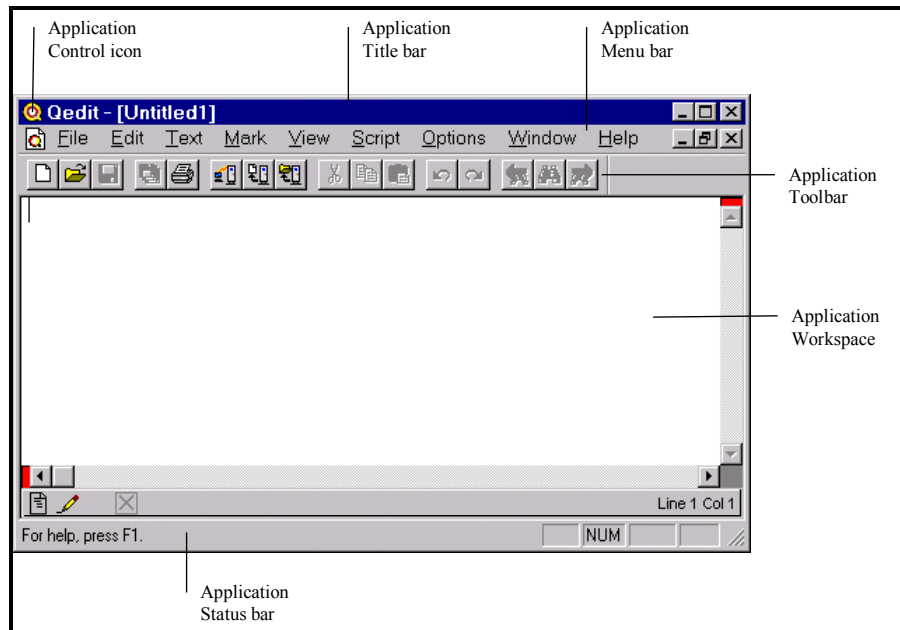


Qedit Startup dialog box

An error does not prevent Qedit from running. However, some features might not be functional. You should correct the problem and restart Qedit.

The Application Window

When you start Qedit, you will see the following application window:



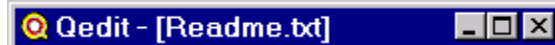
Qedit application window

Application Icon and Control Menu

By clicking the horse icon, which identifies Qedit, you can access the application control menu. This menu contains commands that modify characteristics in the application window, such as its size and location. These standard Windows commands are described in “Controlling the Size and Position of Windows” on page 37.

Application Title Bar

The title bar displays information about Qedit as well as the document you are working on. If files are open, the name of the active document appears on the title bar. Otherwise, only the application name (Qedit) appears on the title bar.



Title bar for a local file

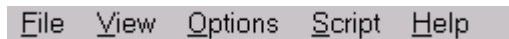
If the active file is on a host, the connection name appears as a prefix of the file name. In the example below, the connection name is Vancouver - MPE and the file is called Cobol4.Pub.Demo.



Title bar for a host file

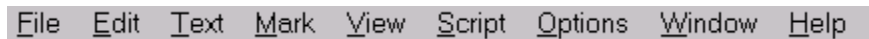
Application Menu Bar

The menu bar displays the names of menu headings, which provide access to a variable number of commands. The number of headings on the menu bar depends on whether a file is open. The default menu bar, when you have no files opened, has the following five (5) menus.



Short menu bar

As soon as you open one file, the menu bar expands to the following nine (9) menus.



Complete menu bar

To display the commands on a menu, click the menu name. To choose a command, select it from the list. Available commands appear in a solid color. Commands that appear dimmed are not available, often because a prerequisite action is necessary. For example, Qedit does not allow you save a file unless there are unsaved changes.

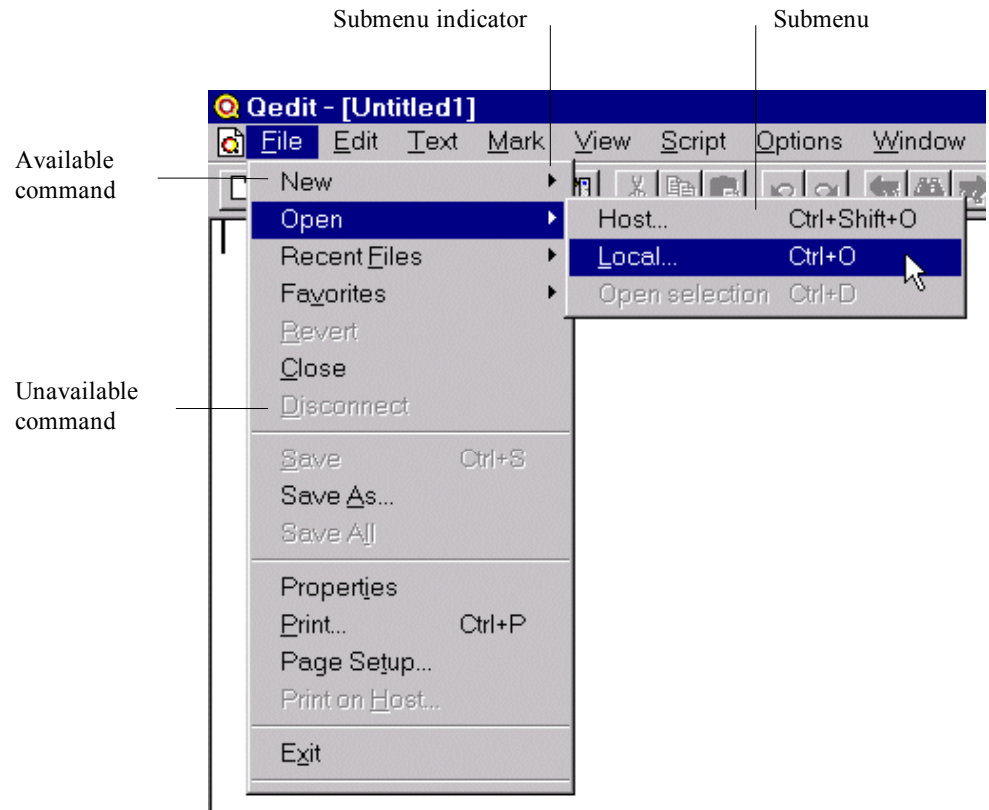
There are times where commands are available on local files but are dimmed when editing host files. This means the command is not supported in the server you currently use. If this happens, you should make sure you have installed the latest version of the server.

Some commands also have a submenu with more choices. These commands can be identified by the small dark arrow on the right.

You can also use keystrokes to access menus and their commands. You can press the ALT key once to select the **File** menu. To go to a different menu, use the cursor keys to move left or right. To see the commands in the current menu, simply press ENTER. You can then use the UP ARROW or DOWN ARROW keys to select a command.

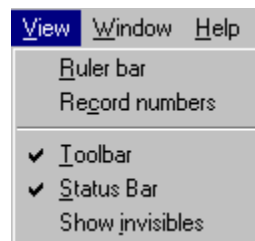
All menus and some commands have an underlined letter, which is called an *access key*. You can combine the ALT key with the access key to quickly display a menu. If a menu name is already selected, just press an access key to move to and display

another menu. If commands that have a defined access key are already displayed on a menu, you can also use the access key to select them.



Menu, commands and submenu

Some menus have checked commands, which use a check mark to show that they are enabled. In the **View** menu below, the check mark indicates whether the command is enabled or disabled. To change the status of a checked command, simply select it.



Checked commands

Application Toolbar

A number of Qedit functions are also available with a single click of the mouse. These functions appear on the toolbar as a series of buttons. If you prefer not to see the toolbar, you can hide it by selecting the **Toolbar** command in the **View** menu. This setting is saved across invocations, which means that if you hide the toolbar, it

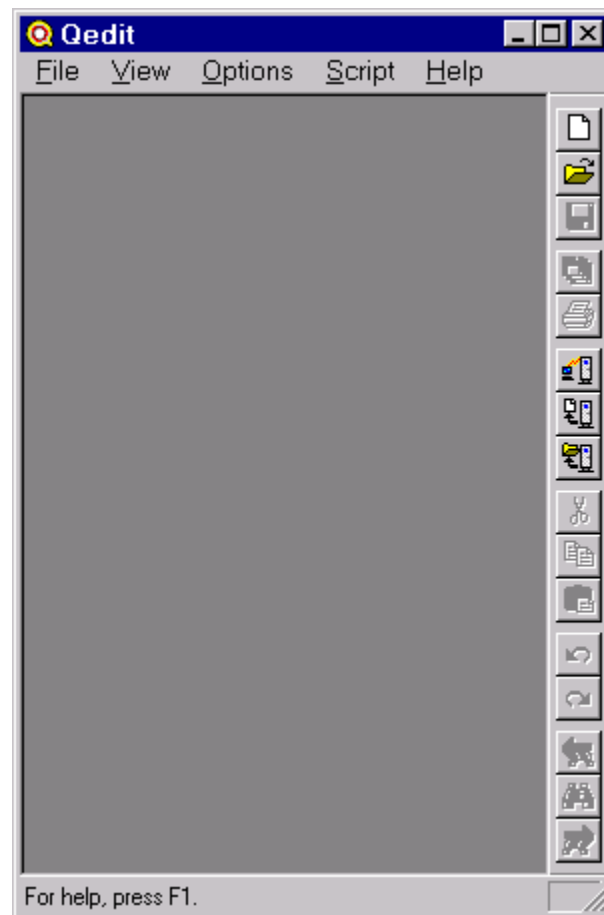
will remain hidden even if you exit and restart Qedit. To display the toolbar again, you have to enable it in the **View** menu.

Changing the Shape and Position of the Toolbar

The default location for the toolbar is underneath the menu bar at the top of the application window. Under Windows 95, it is possible to move the toolbar to other locations on the document window.

For a vertical toolbar, move it completely over to the left or right side of the window.

To move the toolbar, click and hold the mouse on the area between buttons. Then drag the toolbar to the position you want and release it. You can move it left, right or to the bottom of the window, or even display it vertically on either side of the window.



Vertical toolbar on the right side of the window

Turning the Toolbar into a Vertical Column

To shape the toolbar into a vertical column,

1. Place your pointer as far as possible to the left (if you want to move the toolbar to the right) or to the right (if you want to move the toolbar to the left).
2. Drag the toolbar as far as possible to the side that you want until its outline changes to a vertical column.
3. Release the mouse button.

Turning the Toolbar into a Floating Toolbox

You can also resize the toolbar and make it into a rectangle that can be moved around the window. Move the mouse pointer toward one edge of the toolbar. When the pointer changes to a double-sided arrow, click and hold the mouse button and move the cursor in the direction of one of the arrows.



Rectangular toolbar

Note that the position of the toolbar is not saved when you close Qedit. That means every time you run Qedit, the toolbar returns to the top of the application window.

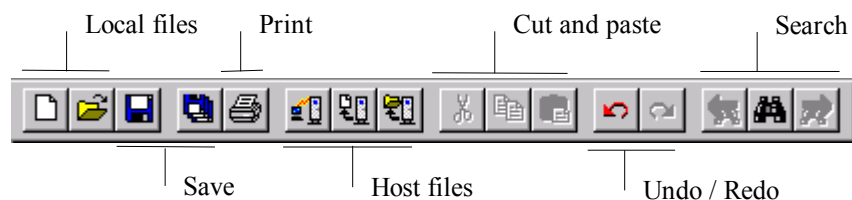
Retrieving a Hidden Toolbar

If your toolbar disappears on Windows 95, there are two ways to retrieve it:

- Move the taskbar somewhere else on the screen and move the toolbar back up. This works only if you have configured the Windows 95 taskbar to **Always be on top**, which lets you move Qedit's toolbar all the way to the bottom of the window and hide it behind the taskbar.
- Shut Qedit and start it again.

Toolbar Buttons

The application toolbar contains the following groups of buttons:



Toolbar buttons and groups

Local Files Tools

The Local files group includes two buttons: the **New** button creates a new file and the **Open** button opens an existing file.



New and Open local files tools

Save Tools



Save tool



Save All tool

The **Save** tool, which appears by itself, can be used for both local and host files. The **Save** button appears dimmed if you have not made any changes to the current file. The button becomes available as soon as you edit the file. If you are editing a Qedit workfile on a host, the **Save** button is never available because changes to these files are posted immediately.

The **Save All** tool works basically the same way as the **Save** tool. The only difference is that **Save All** allows you to save the changes made on all currently opened files. This way, you don't have to save each window. Of course, the downside is it saves all your changes without asking for confirmation. If you want to save only some of the files, you still have to do them one by one.

Print Tool



Print tool

The **Print** tool can be used for both local and host files. You can also control where you want to print the file. A local file can be printed only on a local printer that is attached to your PC. A host file can be printed on a local printer or on a printer connected to the host.

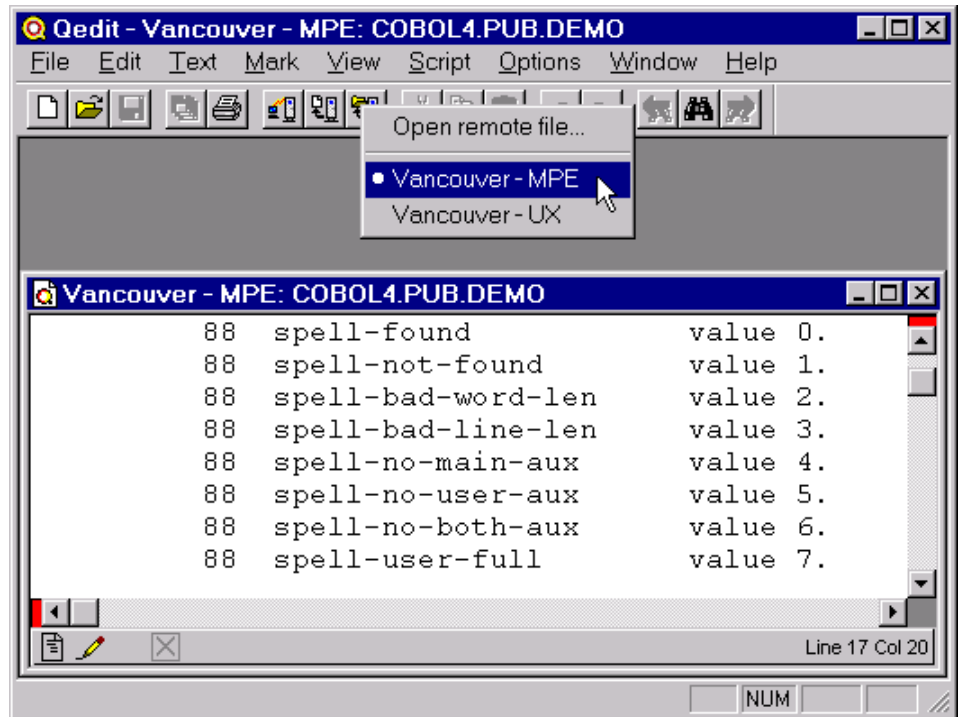
Host Files Tools



Manage connections, Create and Open host file tools

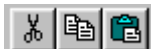
The Host files group includes three buttons: the **Manage Connections** button handles your connections, the **Create host file** button opens a new host file, and the **Open host file** button accesses an existing host file. To invoke one of these operations, simply click the button once. If you click and hold any of these buttons, Qedit displays the primary command followed by a list of available connections. In the list, connections that are currently opened are identified by a bullet.

For example, if you select a connection name under the **Manage connections** button, Qedit automatically displays the **Directory** window for that connection.



Click and hold on the Open host file tool

If you select a connection name under the **Create host file** or **Open host file** buttons, the appropriate dialog box is displayed with the name you selected in the **Connection** text box.



Cut, Copy and Paste tools

You can erase the contents of the Clipboard with the Clipboard Viewer program in the Windows Accessories.

Cut-and-Paste Tools

The cut-and-paste group includes three buttons allowing easy access to the **Cut**, **Copy** and **Paste** functions. If you have not selected any text and have not sent anything to the Clipboard, all three buttons are unavailable and appear dimmed. If you select a block of text, the **Cut** and **Copy** buttons become available.

Once you cut or copy your selection to the Clipboard, the **Paste** button becomes available. It remains available until you cut or copy a new selection, even if you paste the contents. This allows you to copy the same selection to different locations.



Undo and Redo tools

Undo and Redo Tools

The undo/redo group includes two buttons allowing you to reverse your edit operations or re-apply an operation that has been reversed. When you click **Undo**, Qedit removes the changes you have made one at a time, starting from the most recent change. Of course, if you have not made any changes yet, the **Undo** tool is unavailable and appears dimmed.

The **Redo** button is available only if you have used the **Undo** button at least once. It is used to re-apply the operations that have been reversed by an undo operation, starting with the last one.



Find Previous, Find and Find Next tools

Search Tools

The search group includes three buttons allowing you to find strings within the file: **Find previous** locates the previous occurrence of the string, **Find** locates a new string or changes the search options, and **Find Next** locates the next occurrence of the string.

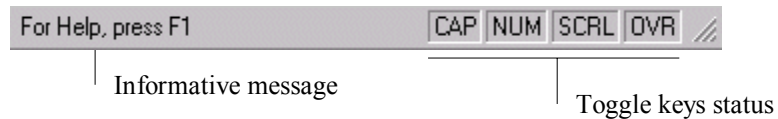
To define the search criteria, select the **Find** button so that you can specify the string and any search options you want to use in the **Find String** dialog box. Once you have used the **Find** button, the **Find previous** and **Find next** buttons become available.

Application Workspace

Qedit uses the application workspace to display all its document windows. The workspace can display multiple documents at the same time or only one document can occupy the whole area. Document windows cannot be moved completely outside the application workspace.

Application Status Bar

The status bar at the bottom of the application displays various indicators. If you prefer not to see the status bar, you can hide it by selecting the **Status Bar** command of the **View** menu. This setting is saved across invocations, which means that if you hide the status bar, it will remain hidden even if you exit and restart Qedit. To display the status bar again, you have to enable it in the **View** menu.



Status bar of the application window

On the left side of the status bar, Qedit displays informative messages as you use certain functions. For example, if you use the **Create local file** button on the toolbar, a short message appears on the status bar explaining what you are about to do.



Status bar while using a tool button

Toggle Keys

There are four functions that you can switch on or off: CAPS LOCK, NUM LOCK, SCROLL LOCK and INSERT. Each function has an indicator box in the Qedit status bar. When a function is disabled, its corresponding indicator box appears blank.



Status bar with all toggle keys turned off

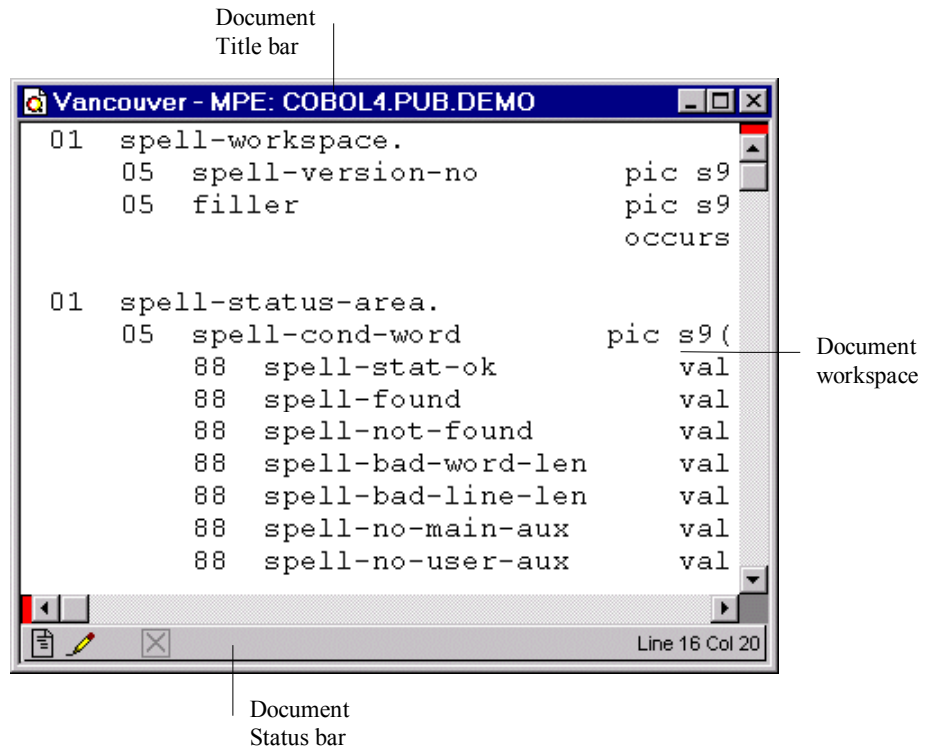
The toggle keys perform the following functions:

- The CAPS LOCK key puts new text in uppercase.
- The numeric keypad on the right side of your keyboard can be used to either enter numbers or move the cursor. The NUM LOCK key makes the numeric keypad enter only numbers.
- In some applications, the SCROLL LOCK key controls the behavior of the insertion point while working on a document. Qedit simply indicates the status of the key. It does not have any effect on Qedit's operations.
- The INSERT key puts Qedit into insert mode, which controls the way you enter new text.

The Document Window

Document windows are sometimes called child windows.

The Qedit application window does not directly access files. Instead, when you open or create a file, Qedit creates another window called a document window. Although each document window has the same layout, they work independently.



Document window

Document Title Bar

The document title bar appears at the top of a document window. It holds three distinct elements:



Document title bar

- Windows control menu

In the example above, the icon on the left side of the title bar gives you access to the document window control menu. This menu allows you to move, re-size, close the current window or switch to another window.

- File name

The middle section of the title bar displays the fully qualified file name. For local files, this is simply the name itself including any extensions (e.g., Readme.txt). For host files, the file name is prefixed by the connection name. In the example above, the connection name was Vancouver - MPE followed by a colon and the file name.

- Windows control tools

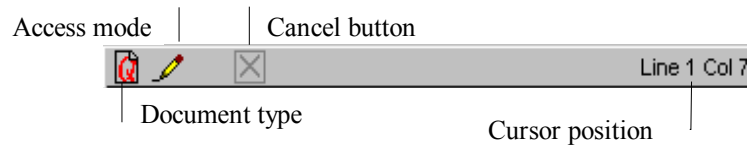
The tools on the right hand side of the title bar are the standard control buttons provided in Windows 95.

Document Workspace

The document workspace is the area in which a file appears within the document window. As you move through the file, Qedit displays only the part of the file you need to see and records any changes you make.

Document Status Bar

The status bar is a dynamic area with a number of indicators and values. It changes with almost every action you perform on the document workspace. Typically, the status bar displays the document type, the access mode, a **Cancel** button and the current cursor position.



Document window status bar

Document Type

The document type indicates whether you are working with a normal text file or a Qedit workfile.



Text file

Regular text files can be found on both local and host files. If the document type icon has black and red lines, it means the file has been modified but has not been saved yet. If the icon has only black lines, it means the file has not been modified since the last save.



Qedit workfile

Qedit workfiles exist only on a host; currently there is no equivalent format for PC files. When you edit a workfile, the document type icon always appears the same. This is because changes in these files are posted directly to the host server. As a result, the **Save** command on the **File** menu *is available* and the **Save** button is not available when you work on a Qedit workfile.

Access Mode



Write access

Qedit displays the type of access you have to a file, write or read-only, by using a pencil icon.



Read-only icon

The pencil icon indicates that you have write access to the file (i.e., you can make changes and save them). This indicator remains on the status bar for the duration of the edit session. If you do not have permission to modify the file, you will see the read-only icon.

Cancel Button

The **Cancel** button is available when Qedit performs operations that are lengthy. When available, you can click the **Cancel** button to stop the operation before it is complete. For example, if you ask Qedit to print the current file, you might see the following status bar:



Cancel button available during print operation

Qedit shows a printing operation under way, and the **Cancel** button is available.

Cursor Position

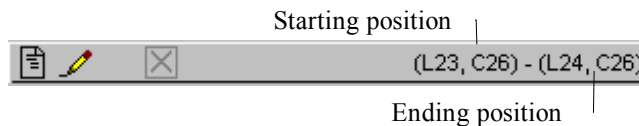
The cursor position is normally a relative record and column coordinate. In the example above, the cursor is currently in the first column of line number 42.

If you select some characters on a single line, the coordinates appear as the starting line, the column numbers and the number of characters selected.



Cursor position with selected text on a single line

If you select a block of text that spans multiple lines, the coordinates appear as the starting line and column, and the ending line and column.

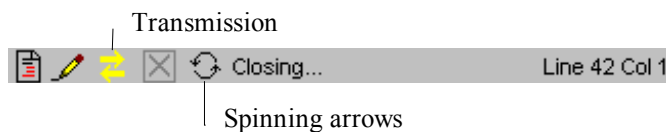


Cursor position with selected text on multiple lines

In this example, the selection starts in column 24 of line 33 and ends in column 57 of line 36. If the selection appears on a single line, the status bar shows the number of characters selected besides the coordinates.

Operation in Progress Indicators

Depending on the operation in progress, Qedit displays progress indicators (e.g., the transmission icon and spinning arrows) when appropriate.



Other optional indicators

For example, the transmission icon indicates the client is waiting for an answer from the server. The icon disappears as soon as the information is received. Normally, you can continue to work while the arrows are yellow. Red arrows indicate there are too many messages in the server queue, in which case Qedit doesn't allow further editing. You will be able to resume your work when the icon becomes yellow again or disappears from the status bar.

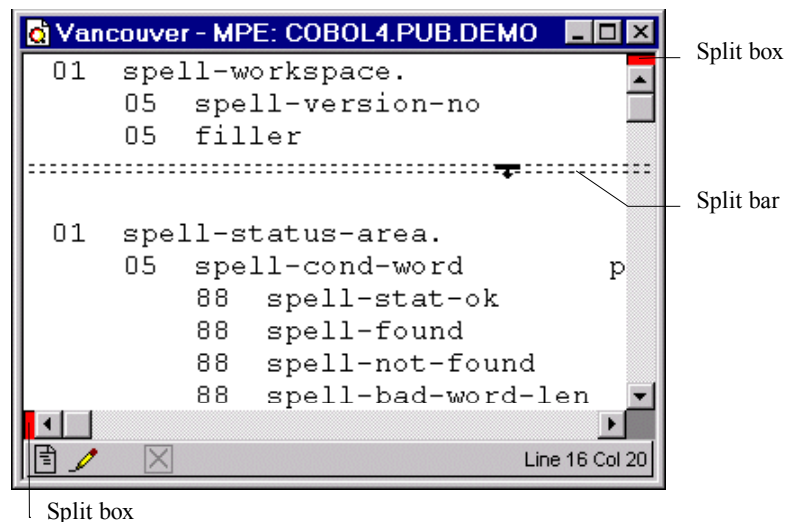
The spinning arrows (they are animated and really appear to be spinning), which indicate an operation in progress, appear whenever Qedit thinks the operation might take some time to complete. Arrows can appear during these operations:

- Opening
- Closing
- Searching
- Working
- Printing
- Pasting
- Replacing

Splitting a Document into Independent Views

The split box appears as a red tab at the top of vertical scroll bars and on the left of horizontal scroll bars.

You can look at two different sections of the same file by using the split view feature of Qedit. If you put your mouse on the split box, the pointer changes to a dark bar with an arrow. When you drag the split bar in the direction indicated by the arrow, you will see two dotted lines dividing your document into two parts. It should look like this:

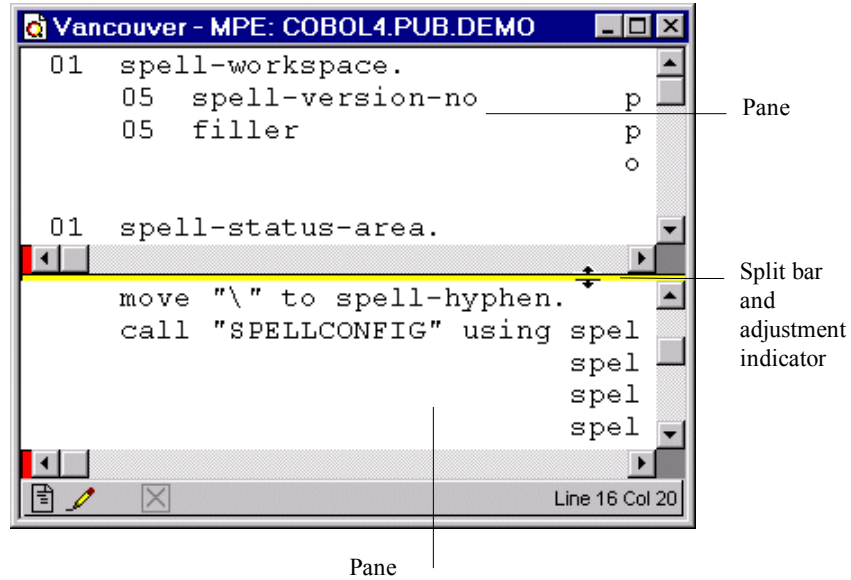


Splitting a window using the split bar

If Qedit cannot split a window at the location you select, you will see a *single* dotted line. You can split the screen only when the split bar becomes solid. Be careful when you move a split bar. If the split bar changes to a dotted line while you are moving it, the split bar will be *removed* (i.e., you will no longer have a split view).

Split bars are displayed as yellow lines between two panes.

When you release the mouse button, if your window is big enough to split, Qedit will split it into two windows that scroll independently. A split bar will delimit the two panes, each with its own set of scroll bars. If you have enough room, you can split panes into more panes. You can adjust the size of each pane by moving the split bar.



Adjusting the pane size using the split bar

Alt+W,S then H for Horizontal or V for Vertical

Shortcut: double-click the split box.

You can also use the **Split window** command in the **Window** menu. This command has two (2) subcommands (**Horizontal** and **Vertical**), which split the active window (or pane) in half in the appropriate direction. If you want to quickly split a window or a pane in half, double-click the split box.

Collapsing Multiple Views into One

Shortcut: double-click the split bar.

To collapse two views of a file into one, drag the yellow split bar to the edge of the window and release it.

A Windows Primer

Windows Standard Functions

Qedit for Windows conforms to many of the standard functions in Microsoft Windows. If you haven't had much experience with Windows, this chapter covers many of the Windows features that Qedit for Windows uses. These features include using the mouse to select text, minimizing and maximizing document windows, using toolbar buttons, and generally enjoying the many time-saving shortcuts a Windows interface provides.

For a complete description of Microsoft Windows, consult your Windows manual.

Switching Between Applications

Without a mouse, you can still switch to another Windows application. Hold down the ALT key and press TAB. This brings up an icon list of all your running applications. If you let go of the ALT key, you will switch back to the last application you were in. While keeping the ALT key down, you can continue to press TAB until you reach the application you want, or go back to Qedit. Release the ALT key once you have selected the application you want. Use the same procedure to go back to Qedit.

Controlling the Size and Position of Windows

Moving a Window Around on the Screen

To move a window, first point to the title bar at the top of your Qedit window, then click and hold the mouse button. You can now drag your window anywhere on the screen.

Resizing a Window

To resize a window, put your mouse on any of the edges or corners of your Qedit window. When your cursor changes into a double-ended arrow, drag it to make the window larger or smaller.

Unless the text font you are using is really large, the minimum window size is set by the status bar. Qedit for Windows does not allow the status bar to be narrower than the widest message or indicator.



Minimize, Maximize and Close buttons on Windows 95.

Minimizing a Window

In the upper right corner of most windows are some control buttons that allow you to minimize the window. The **Minimize** button reduces the program to an item on the taskbar in Windows 95.

To restore the window, click the minimized icon and select **Restore** from the menu, or double-click the icon. To restore a program from the Windows 95 taskbar, just click it once.

Maximizing a Window



Minimize, Restore and Close buttons on Windows 95.

In the upper right corner of most windows is a **Maximize** button that enlarges a window so that it occupies the whole screen. If you want a large Qedit window, you need to maximize both Qedit itself as well as the document window within it. When a window is maximized, the **Maximize** button changes to a **Restore** button, which returns the window to its previous size.

If you want to easily maximize or restore a window, double-click the title bar.

Shortcut: To easily maximize or restore a window, double-click the title bar.

Arranging Windows into Tiles

To tile documents horizontally, use ALT+W followed by H. Currently opened windows will be arranged to use the width of the application workspace. The windows will then appear one on top of the other.

To tile documents vertically, use ALT+W followed by V. Currently opened windows will be arranged to use the height of the application workspace. The windows will then appear side by side.

You can quickly switch between files by moving your mouse to another document. See also “Tiling Document Windows” on page 182.

Shortcut: ALT+W,H

Shortcut: ALT+W, V

Closing a Window

This is a standard Windows 95 function. In the upper right corner of most windows, there is an X button which closes the window. With earlier versions of Windows, you click the window control button with the straight line bar in the upper left corner, then select **Close** from the resulting menu.

Mouse Operations

Almost all computers have a graphical pointing device, which can be a pen, a mouse, a trackball or a stylus. Pointing devices normally have buttons on or close to them. If a device has multiple buttons, usually one is identified as the primary button.

The pointing device for Qedit for Windows is a mouse, which mostly behaves the same as mouse devices in other Windows applications. Qedit does, however, have two special mouse operations: dragging the split box on the scroll bar to split a window (see “Splitting a Document into Independent Views” on page 35) and clicking and holding the toolbar host buttons to get a list of connections (see “Shortcut Menus on Toolbar Buttons” on page 44).

Some models of mouse devices, such as the Microsoft Intellimouse, have a wheel between the buttons. This wheel lets you maneuver through a file easily. Qedit gives you the ability to control the scrolling speed of an intelligent mouse. Simply adjust the **Mousewheel Acceleration** slider in the **Preferences** dialog box. See “Using the Intelligent Mouse” on page 136 for more details.

Using Mouse Buttons

Qedit for Windows recognizes a number of standard ways to use the mouse:

- Click
This means a single click of the left mouse button, which is usually the primary button.
- Double- or Triple-Click
This means two or three left mouse clicks in very rapid succession on the specified object or location on the screen. The speed at which you have to click, (i.e., the time between the clicks) is configurable with the **Mouse** tool under the **Windows Control Panel**.
- Click and Hold
This means a single click of the left mouse button and holding it down.
- Drag
Dragging is used to select text. Click and hold the mouse across the text you want to select. When you reach the last character, release the mouse. The text you selected appears highlighted.
- Drag and Drop
This is used to move objects on the screen. Click and hold the left mouse button on an object and move it to a new position on the screen. For example, you can move document windows on your screen.

Although the mouse is very handy in Qedit for Windows, you can use the keyboard for most tasks in Qedit. For example, you open a file and want to edit a word near the bottom of the page. Press the DOWN ARROW key until it starts to repeat, then let up just in time. If you miss, go up or down a line. Press the RIGHT ARROW key until you reach the desired column. If you overshoot, you are in column one of the next line. If you pick the wrong line and it is shorter than your target line you have to go right, then down, then right again. This is an error-prone task. It can be more efficient to use the mouse to move the insertion point to the word you want.

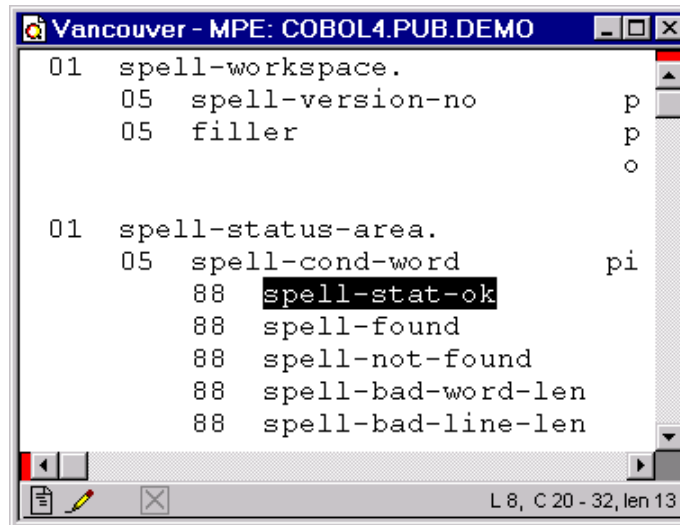
TIP: If your mouse seems sticky, it is probably dirty. Mouse devices pick up lint and need regular cleaning. The cleaning process is relatively simple: twist off the cover on the ball, let the ball fall out, and scrape the lint off the three rollers with a finger nail or sharp object. Then replace the cover.

Selecting Text with a Mouse

Selecting (or highlighting) text is a basic operation of Qedit. Without it, your editing is very limited. When text is highlighted, it appears in inverse video and is available for further processing. There are several ways to select text with a mouse.

Dragging the mouse over the text is one way to highlight. To select *spell-stat-ok* in the example below,

1. Put your mouse just in front of the letter *s* of the word *spell* and hold the left mouse button down.



Selected text

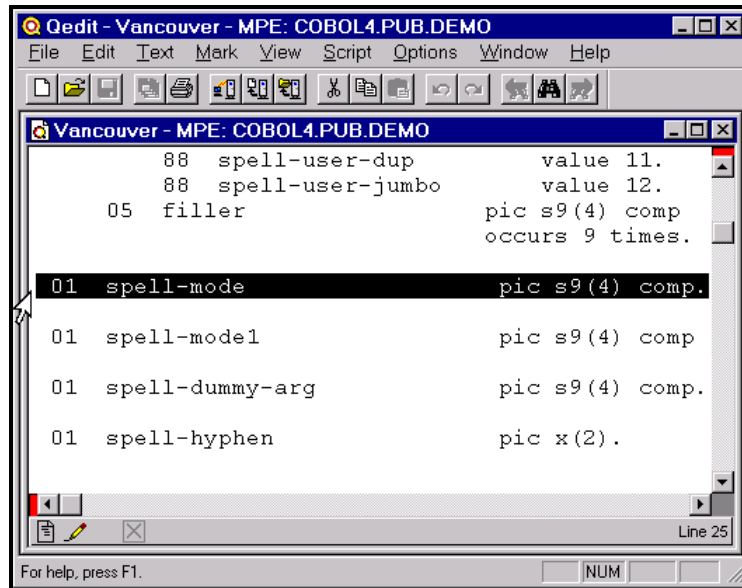
2. Carefully move the mouse to the right, swiping over the rest of the phrase. As you move, the intervening characters will be highlighted.
3. When the *k* in *ok* is highlighted, release your mouse button. The selection you highlighted should look like the example above.

If you lift the left mouse button before you have finished dragging it, you will not select the whole phrase. If you move the mouse too far to the right, simply move it back.

You can also select text by clicking the mouse:

- Double-click anywhere on a word and Qedit selects the whole word automatically. If you start dragging the mouse after a double-click, Qedit continues to select whole words. You can configure Qedit to select spaces following the word using the option on the General tab of the Preferences dialog box.
If the **Word selection includes trailing spaces on double-click** option is enabled, Qedit selects the current word along with all spaces, if any, that immediately follows. If the option is disabled (default), Qedit selects the word only.
- Click three times to easily select a whole line. If you start dragging the mouse after a triple-click, Qedit continues to select whole lines.

You can quickly select complete lines by moving the mouse pointer to the left margin, where it changes to a right-pointing arrow. Click once and Qedit automatically selects the whole line. Click and drag to select more lines.



Right-pointing arrow for easy line selection

Complete lines are usually highlighted all the way to the right margin and include end-of-line characters, which are invisible by default. If you paste the complete lines somewhere in the file, they will also be on a separate line. If the end-of-line character has not been selected, the pasted text will appear on the same line as the existing text.

There is only one exception to this rule. The last line of the file does not have an end-of-line character. If you highlight the complete last line, it appears just like any other complete line (i.e., the highlight goes all the way to the right margin). However, if you paste it somewhere else, it will not be put on a separate line.

If you press DELETE, the highlighted text is removed and the lines on either side are joined. For more details, see “Selecting Text” on page 92. To cancel a selection, click anywhere else in the document window.

Selecting More Text

If you click and drag the mouse up or down to a different line, Qedit selects the rest of the current line and part of the next line up to the mouse pointer.

Extending a Selection

If you want to select a lot of text, you can extend your selection using the SHIFT key.

1. Place your cursor at the beginning (or at the end) of the text you want to select. If you want to make sure the cursor remains in place, select part of the text immediately.
2. Move the cursor to the other end.
3. Press and hold the SHIFT key.
4. Click at the location where you want the selection to end. Qedit automatically selects anything in between.

This technique can be used in other circumstances. For example, you can use it to open more than one local file in a single operation.

If you want to select one or more files that appear next to each other,

1. Click the first file.
2. Hold the SHIFT key.
3. Click the last file you want opened. All files between your first and last selection will be highlighted.
4. Click **Open**.

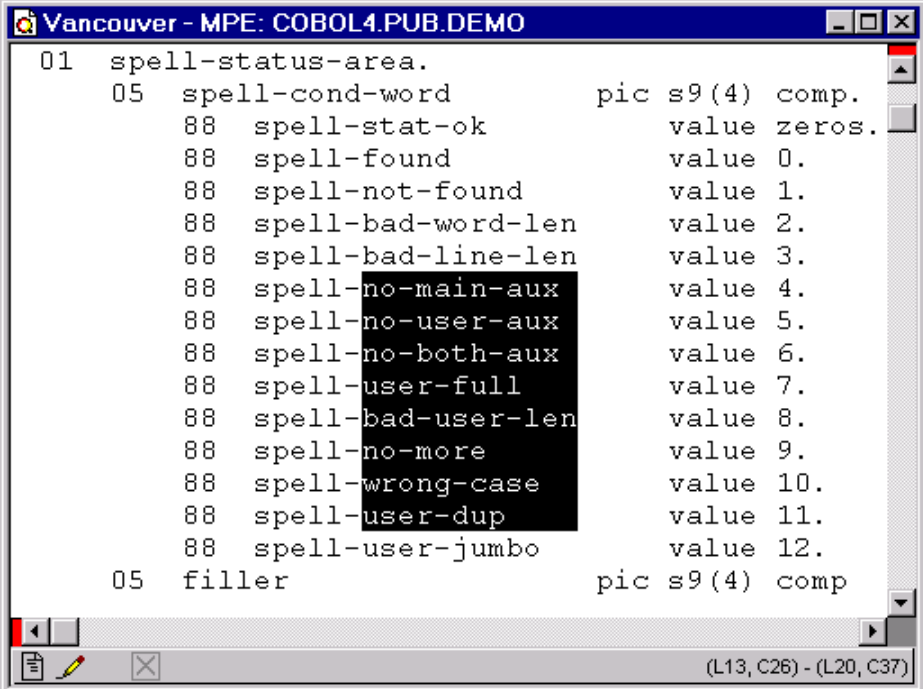
You can also select files that are not next to each other using the CTRL key.

1. Select the first file.
2. Hold the CTRL key.
3. Select another file that is not directly adjacent to the first one.
4. Both files are highlighted. Files appearing between these two are not selected.

You can also combine the CTRL and SHIFT keys to select even more files in the listing. By releasing the SHIFT key and pressing the CTRL key, you can skip the files you do not want to open.

Rectangular Selections

If you want to work on text in specific columns on multiple lines, you create something called rectangular selection. Simply hold the CTRL key while clicking and dragging the mouse. The selection is then limited to the leftmost and rightmost columns in the selection as shown here:



```
01 spell-status-area.  
   05 spell-cond-word          pic s9(4) comp.  
      88 spell-stat-ok         value zeros.  
      88 spell-found           value 0.  
      88 spell-not-found       value 1.  
      88 spell-bad-word-len    value 2.  
      88 spell-bad-line-len    value 3.  
      88 spell-no-main-aux     value 4.  
      88 spell-no-user-aux     value 5.  
      88 spell-no-both-aux     value 6.  
      88 spell-user-full       value 7.  
      88 spell-bad-user-len    value 8.  
      88 spell-no-more         value 9.  
      88 spell-wrong-case      value 10.  
      88 spell-user-dup        value 11.  
      88 spell-user-jumbo      value 12.  
   05 filler                   pic s9(4) comp
```

Rectangular selection

Rectangular selections can be defined with the mouse only. There is no keyboard equivalent.

Selecting a Menu Command

The line of words across the top of the Qedit window (**File, Edit, View**, etc.) is called the menu bar. Each word is the name of a menu, which you can click to access a list of commands. Commands that are dimmed are not available at the time, usually because a prior operation is necessary.

To open a file, for example, first click the **File** menu. When a list of commands appears, you can select the **Open** command, which presents you with more options (**Host, Local, Selection**).

Selecting Toolbar Buttons

Below the menu bar is a toolbar with various small square buttons.



Qedit toolbar

If you are using the 32-bit edition of Qedit, you can find out what a button does by putting your cursor over it and waiting. Do not click. A toolbar tip will shortly appear to describe its function. To activate the function, click it once. If the button appears dimmed, it means that you cannot do that function at this time, usually because you have not selected anything to do the operation on.

Selecting from a List

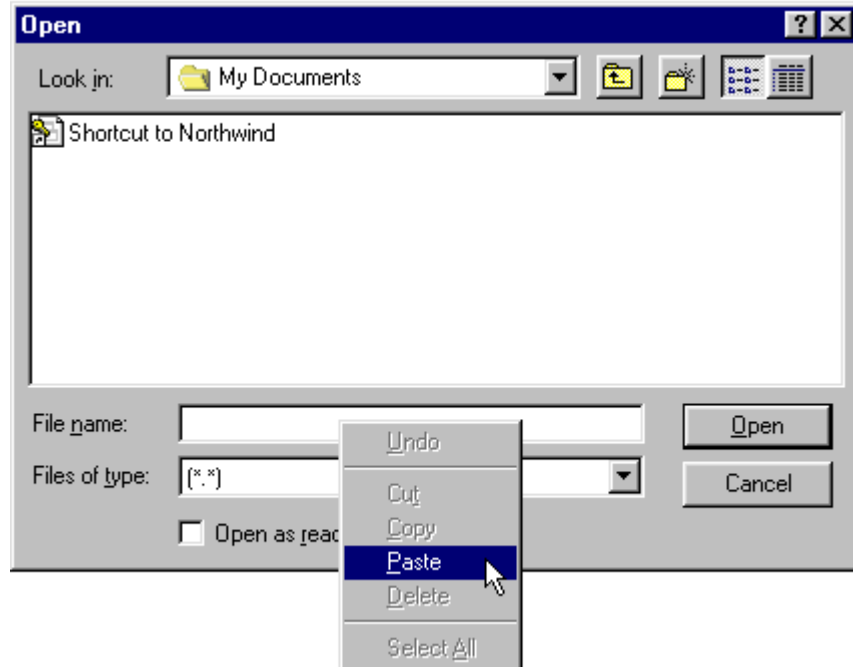
You can make a selection from a file list by clicking the arrow at the end of the list. This expands the file list box and displays all the files currently saved in the list. Use the mouse to point to the file you want and click to select it. You can also use the DOWN ARROW and UP ARROW keys to display the previous or next file name in the list.

Shortcut Menus in Windows 95

Shortcut menus were introduced in Windows 95. They are context-sensitive menus whose contents vary according to where the mouse points. You access shortcut menus by right-clicking the mouse.

Shortcut Menu on Text Boxes

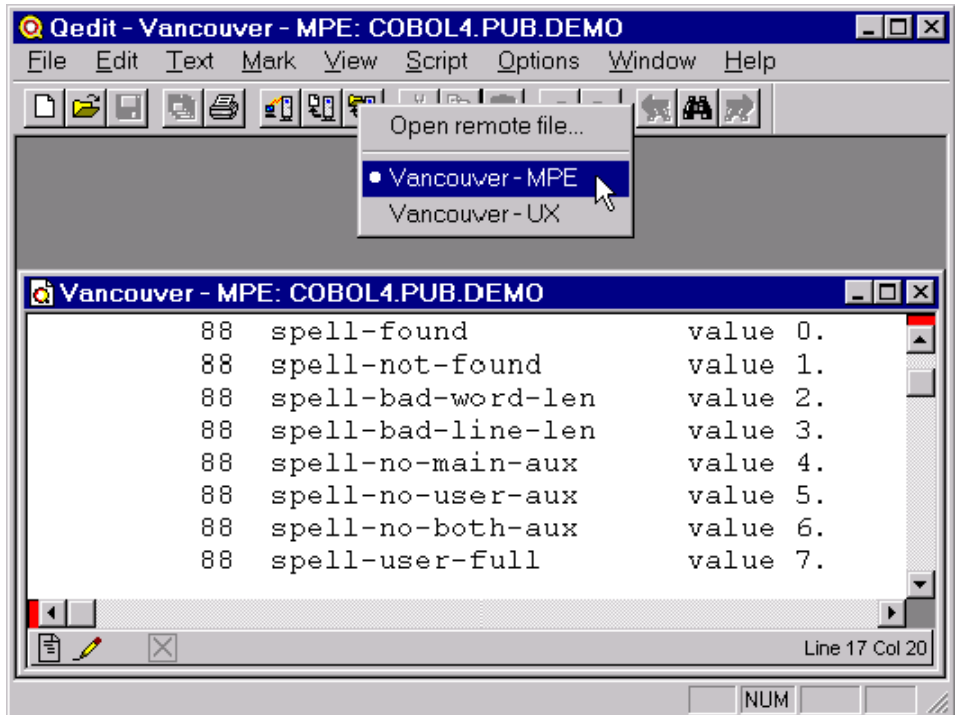
Typically, these menus give you easy access to Clipboard functions such as **Cut** and **Paste**. In Qedit, shortcut menus are available on some text boxes. For example, if you have copied a file name to the Clipboard and you want to use it to open the file, display the **Open** dialog box and right-click in the **File name** text box. Once the shortcut menu is displayed, simply select **Paste**.



Right-click a text box to display a shortcut menu

Shortcut Menus on Toolbar Buttons

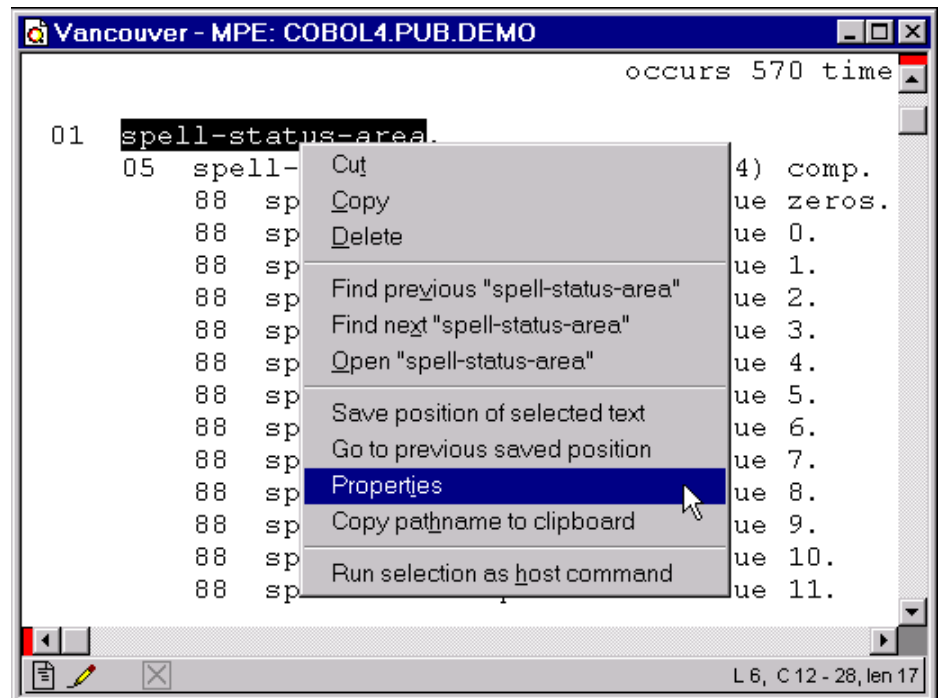
Qedit's toolbar has one special shortcut function. If you press one of the three host buttons and keep it down for a few seconds, a menu of host options will appear. Just select the host name on which you want to open a file, open a new file, or search the directory. To cancel the menu, click somewhere else on the screen.



Click and hold the Open host file button

Shortcut Menu on Document Workspace

A shortcut menu is also available if you right-click in the document workspace area. The list of commands in the menu varies with the current conditions.



Shortcut menu for text manipulation and to access file properties

These options are not available if the selected text spans more than one line.

In the example above, the commands in the top part of the shortcut menu (**Cut**, **Copy**, **Paste** and **Delete**) are available if you have selected some text. They provide easy access to common editing and Clipboard functions. The **Paste** command is only available if there is something in the Clipboard.

If there is something in the Clipboard but you have not selected anything, you might also get a **Paste** command.

The commands in the middle part of the menu (**Find previous**, **Find next** and **Open**) are available if the insertion point is immediately in front of or anywhere on a word or if you have selected some text. You do not have to select a word. Qedit will automatically insert the selection in these commands for easy searches or file opens. If you have selected some text, the selection is inserted in the commands.

If you select one of the **Find** commands, Qedit uses the search options used on the last **Find** operation.

The **Save position of insertion point** command saves the current cursor position on the caret stack. If you have selected some text, the command reads **Save position of selected text**. The **Go to previous saved position** command jumps to the location last saved on the caret stack.

The next command in the menu, **Properties**, gives you direct access to the **File properties** dialog box so that you can change document attributes. This dialog box can also be accessed using the **Properties** command in the **File** menu.

The last command, **Copy pathname to clipboard**, puts the fully-qualified name of the active document in the Clipboard. For an MPE host file, this would be in the form of

`<file>.<group>.<account>`

If the file is in the POSIX namespace of an MPE host or the file is on a UNIX host, the format is:

`<absolute pathname>/<file>`

If the file is local, the format is

`<drive:><absolute pathname>\<file>`

If you are working on a host file, there is a command to execute the file as host commands. If there is no active selection, the command is **Run file as host commands**. If there is a selection, the command is **Run selection as host commands**.

Send To Function in Windows 95

In Windows 95, when you right-click a file name in Windows Explorer or in a file list, such as an **Open** dialog box, you will see a list of context-sensitive options available for that file. One of the options is **Send To**, which permits you to send the file to a diskette, a folder, a program, etc. You can add more destinations to the list of **Send To** targets, including Qedit for Windows. When you send a file to Qedit in this way, Qedit automatically starts and opens the file.

Adding Qedit as a Send To Target

This procedure is usually included when Windows 95 is installed. On your PC, however, the drive letter and the SendTo directory location can be different.



*Qedit for Windows
shortcut icon*

1. In Windows Explorer, go to the C:\Windows\Start Menu\Programs\Robelle directory.
2. Select the Qedit for Windows shortcut icon.
3. Copy the shortcut to the Clipboard by using CTRL+R or the **Edit** menu.
4. Go to the C:\Windows\SendTo directory and paste the shortcut into it by using CTRL+V or the **Edit** menu.

Note that Qedit has difficulty handling long file names passed via the SendTo feature. If you do pass a long file name, it is converted to a short DOS name (8 characters, a period and a 3-character extension). For example,

```
Diary19980811.html
```

would be passed as

```
Diary1~1.htm
```

If you edit this file and save it, Qedit deletes the file with the long file name and replaces it with the new short file name.

Registering File Types in Windows 95

In Windows 95, you can register file types and associate them with an application. This allows you to specify one application to open a file, another application to print it and so on.

You can register Qedit as the default application for certain file types. For optimal use, make sure these files are text only. You should also make sure you are not interfering with the proper operation of other applications. In other words, make sure you are familiar with a file type and understand the implications of registering it with another application before you proceed.

Registering Text Files

For example, to register Qedit with Text documents (i.e., files with a .txt file extension),

1. Go in Windows Explorer.
2. Select the **Options** command from the **View** menu.
3. Select the **File Types** tab.
4. Find the **Text Document** file type entry. If the file type you need does not exist, you have to create it using the **New Type** command. In this case, you need to know more information about the files you want to register.
5. Select **Edit**.
6. From the **Actions** list, select the one you want associated with Qedit.
7. Select **Edit** again.

8. Enter the fully qualified program file name for Qedit in the **Application used to perform action text box**. Qedit is usually installed as `\Robelle\Qedit\Qwin32.exe`.
9. Select **OK** to close the **Editing Action for type** window.
10. Select **Close** to close the **Edit file type** window.
11. Select **Close** again to close the **Options** window.

Registering Qedit Scripting Language Files

If you are using scripts written in the Qedit Scripting Language (QSL), you should also register the `.qsl` extension.

1. Go in Windows Explorer.
2. Select the **Options** command from the **View** menu.
3. Select the **File Types** tab.
4. Select the **New Type** command.
5. In the **Description of Type** text box, enter `Qedit Scripting Language`.
6. In the **Associated extension** text box, enter `.qsl`.
7. Select **New** to add an item to the **Actions** list.
8. In the **New Action** dialog box, enter `Open` in the **Action** field.
9. In the **Application Used to Perform Action** field, enter the Qedit program name e.g. `c:\robelle\qedit\qwin32.exe` or use the **Browse** button to find it.
10. Repeat steps 7 to 9 using `edit` as the action.
11. Select **OK** or **Close** until you're back at Windows Explorer.

Drag-and-drop File Open

Windows offers a way to quickly open a file from an application. Assuming you can see the file and the application, you can use one of the following methods.

Drop on Desktop Icon

In this situation, let's assume the Qedit shortcut icon appears on the desktop.



Dropping a file on the Qedit shortcut icon

1. Click on the file you want to open. You would typically use Windows Explorer to locate the file.
2. Drag it over to the Qedit shortcut icon. The pointer should change to an arrow with a small plus sign enclosed in a square.
3. As the pointer hovers above the icon, the icon is dimmed. At that point, drop the file on the icon i.e. release the mouse button
4. This operation automatically launches Qedit and opens the file.

Drop on Qedit Application Window

If Qedit is already running and the application window is visible, do:

1. Click on the file you want to open. You would typically use Windows Explorer to locate the file.
2. Drag it over to the Qedit application window. The pointer should change to a dimmed outlined square with a small plus sign enclosed in a square.
3. At that point, drop the file on the Qedit window i.e. release the mouse button.
4. Qedit opens the file automatically.

Drop on Taskbar Button

If Qedit is running but its application window is not immediately visible, it might have been minimized. It should appear as a button on the Windows taskbar. In that case, you should:

1. Click on the file you want to open. You would typically use Windows Explorer to locate the file.
2. Drag it over to the Qedit button on the Windows taskbar. The pointer should change to a dimmed document icon.
3. Leave the pointer over the button for a few seconds, Windows should restore the application window.
4. Move the pointer to the application window.
5. At that point, drop the file i.e. release the mouse button.
6. Qedit opens the file automatically.

Special and International Characters

If you do not have a keyboard for languages other than English and you want to insert special characters into your document, Windows 95 has a cumbersome way to do this.

1. Run the **Character Map** program from the **Accessories** group.
2. Select a font from the list. This should be the same font that you defined using the **Properties** command of the **File** menu.
3. For each character you want, either double-click the character or click **Copy**. This automatically adds the character to the **Characters to copy** box.
4. Go back to your Qedit document and paste the copied characters.

Long File Names

Older versions of Windows only allow eight characters for a file name followed by a 3-character extension (e.g., Readme.txt). This limit has been removed in Windows 95 and NT.

The 16-bit version of Qedit, if you are still using it, can only handle files that follow the old convention. If you use this version to edit files on an NT server, for example, you might see some strange behavior. You will be able to open these files, but long

file names will appear different. For example, a file called Ourproducts.html might show up as ourpro~1.htm. Also note the difference in the case used. The original uses upper and lowercase letters whereas the new name uses only lowercase.

If you modify this file and save it, the old name will be replaced by the new file name ourpro~1.htm. If you have to work with files that have long names, you should use the 32-bit version of Qedit.

Connecting to a Host or Network

What do you need to create a connection?

Qedit for Windows requires a client and a server. The client is a program that runs on your PC. The server is a program that runs on one or more of your host computers (HP e3000 running MPE/iX or HP 9000 running HP-UX). The Qedit for Windows server is actually the native-mode Qedit program with a version greater than 4.6, enabled for server mode and run in a special way. Qedit's server mode is an option on host-based Qedit, so you have to talk to someone at Robelle Solutions Technology Inc. to enable it.

On most MPE host computers, you do not have to start the server program because Qedit for Windows takes care of everything for you. If you have special network security devices like firewalls, you might have to use Qedit's firewall protocol. Refer to "MPE Host Connections" on page 211 or the appropriate host-based Qedit manual for complete instructions.

On UNIX hosts, you need to start a listener process. Refer to "HP-UX Host Connections" on page 216 or the appropriate host-based Qedit manual for complete instructions.

A connection is created between the client and the server when you first attempt to access a file or directory on that connection, or when you first run Qedit for Windows if you selected **Login at startup** selected for that connection. It remains connected even though you close all the files on that connection (default), until you either exit from Qedit for Windows or use the **Disconnect** command in the **File** menu.

Connecting Through a Network

In Qedit for Windows, a network is a group of two or more computers that can exchange packets using the TCP/IP protocol. There are other types of networks, such as a NOVELL NetWare network, or a NetBEUI network, but Qedit for Windows cannot run on these networks. The Internet is an example of a Qedit for Windows network. A LAN with TCP/IP enabled is another.

The active computers on the Qedit for Windows network can consist of MPE and HP-UX servers from Hewlett-Packard, plus PCs running Windows 95 or Windows NT (3.51 or 4.0). The Qedit for Windows client software runs on the PC, and the Qedit for Windows server software runs on MPE/iX and HP-UX hosts. A Qedit for

Windows network can also consist of several LANs connected together via gateways to the Internet.

Accessing a File Server

File servers are different from TCP/IP servers running the Qedit for Windows software. A file server can allow you to access files on common disk drives as if the shared disk drive were actually plugged into your own PC. This works best when the file server and your PC are both on the same LAN.

You do not have to create a Qedit for Windows connection to edit files on PC LANs. Any file that you can access from other Windows applications can be edited by Qedit for Windows.

Creating a Connection

A connection is defined as a connection name, a host name, a user login name for UNIX or a “hello” string for MPE, and the passwords needed to access that host. You can create two types of connections: a temporary connection and a saved connection. Saved connections can be accessed quickly because they have already been defined. Temporary connections, convenient for one time use, cease to exist when you close Qedit.

Creating a Temporary Connection

Temporary connections exist as long as you stay in Qedit for Windows. This is a very useful feature for a corporate site with hundreds of systems that are accessed occasionally.

To create a temporary connection,

1. Select the **File** menu, and click **Host** in either the **New** or **Open** commands.
2. When the **New Host File** or the **Open Host File** dialog box appears, click **Create...**
3. When the **Connection List** dialog box appears, select the **Temporary** option.
4. Define the characteristics of your connection.

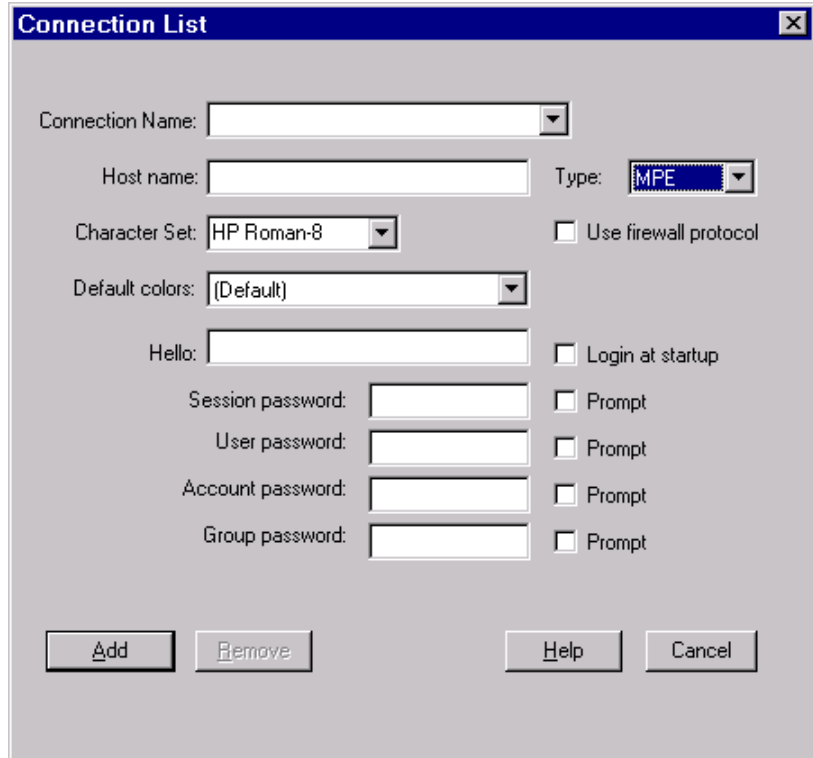
Temporary connections are not added to the connection list.

Creating a Saved Connection

If you need to use a connection repeatedly, you can create it as a saved connection. Once you have created a saved connection, it is accessible through your connection list until you remove it.

To create a saved connection,

1. Select the **Connections** command of the **Options** menu.



Connection List dialog box for MPE hosts

2. When the **Connection List** dialog box appears, enter a **Connection Name**. We recommend that you choose something meaningful to you or anybody else who works on your PC. Examples are:

- the name of your system (e.g., sylvia)
- the system plus your user name (e.g., sylvia - fred)
- the function (e.g., purchase orders)
- something totally irrelevant (e.g., the sky is blue)

The **Connection Name** can contain any combination of letters and numbers, except these characters: the colon (:), parentheses (), and nonprinting characters.

3. Enter the **Host Name** or IP address. You will need the name of your host system in Internet format (e.g., Sylvia.Xyzzy.Com) or the host's exact IP address (e.g., 128.11.200.77).

NOTE: Do not enter leading zeros in any part of an IP address. A leading zero indicates that the value is specified in octal notation. In the example above, if you had entered 128.011.200.077, the IP address would have been interpreted as 128.9.200.63.

4. Select the host type (MPE or UNIX) and define your host login and password(s). Each host type has a different set of labels for defining the login and passwords.

For details, see “Defining an MPE Host Connection” and “Defining an HP-UX Host Connection” below.

5. Although the default character set is HP Roman 8, Qedit gives you other choices in the **Character set** list box. If you are using HP's Native Language Support

subsystem, you can select one of the following: a 7-bit national character replacement set, the new Roman8-1999 (Roman9) or the ISO8859 character set.

The Roman9 character set is identical to Roman8 except for a single character: decimal code 186 now represents the European currency (Euro) symbol. The ISO8859 character set corresponds to Windows Code Page 1252, which allows you to create host files with the Windows character set. These files can then be either transferred to a PC or used directly by PC programs without translating the character set.

6. All new connections are created with the Qedit default color scheme, as indicated by the word (Default) in the **Default Colors** list box. You can select a different color scheme off the list. See "Setting the Default Color Scheme" on page 121 to add your own schemes or change existing ones. Any file opened or created on that connection will use the selected color scheme unless you have selected a different scheme for individual files.

The new default color scheme is only going to take effect when the connection is closed. If it's already opened, you should close it and re-open it.

7. Enter the **Login** or **Hello** string.
8. When the **Add** button appears, enter your password. If your system requires one or more passwords, enter them in the appropriate text boxes. Passwords are saved in the connection template file.

When the prompt option is selected, the corresponding text box becomes unavailable and Qedit will prompt you for a password every time it tries to establish this connection. If you prefer to be prompted for a password, you can choose the **Prompt** option.

9. Click **Add** to create the connection and to save it in your connection list.

Now you can use this saved connection to access your host system. You can create as many connections as you need. Remember, each connection is associated with a specific user login, so you might need several connections to a single host if you need to log in as different users.

Defining an MPE Host Connection

To select the **MPE** option in the **Connection List** dialog box, your HP e3000 host must run the MPE/iX operating system with the Qedit server software installed. Qedit server does not support Classic HP e3000s that run MPE/V.

MPE connections must include the following information:

- **Use firewall protocol**

In most situations, you do not have to do anything special to connect to an HP e3000 host. However, some companies have installed network security devices such as firewalls to restrict access even further. Depending on the installation, these devices might prevent Qedit from communicating to the server using the default MPE connection protocol.

Qedit offers an alternate connection scheme that should solve this problem while satisfying the customer's security needs. For these connections, you have to enable **Use firewall protocol**. Remember that this protocol requires additional installation on the host. Refer to "MPE Host Connections" on page 211 or the appropriate host-based Qedit manual for complete instructions.

- **Hello**

Enter your normal hello command into **Hello** text box, but do not include the word “hello.” Each new session on MPE has its own hello string. A “hello” string can contain up to four elements with the following syntax:

```
<session>, <user>.<account>, <group>
```

Session and group names are optional.

- **Session and Group Passwords**

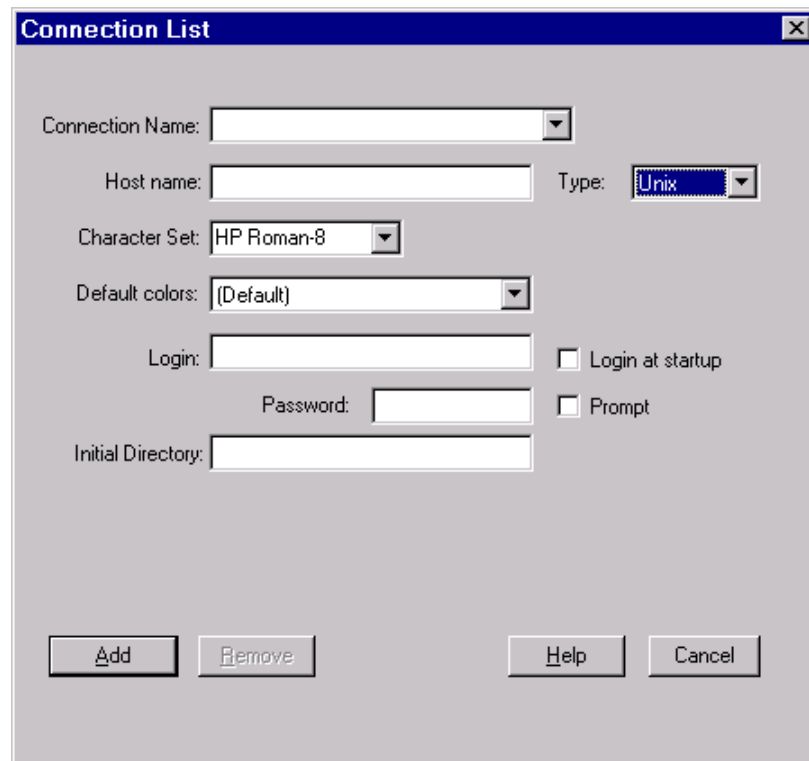
Each element name can have a different password for login information, giving each hello string up to four passwords: Session, User, Account and Group. For example, these are valid commands:

```
ralph.purch,entry  
ordentry,clerk.distrib  
manager.payables
```

Note that the **Session password** is only used with third-party security packages such as VESOFT’s Security/3000.

Defining an HP-UX Host Connection

If you select the **UNIX** option, your HP 9000 host must run the HP-UX operating system, a flavor of UNIX.



Connection List dialog box for UNIX hosts

- **Login**

This is a single user name. It determines your home directory and which files you have permission to browse and edit on the host system. Remember that both UNIX user names and passwords are case sensitive.

- **Password**

You can enter only one password because Qedit does not support multilevel passwords. If your UNIX system uses multilevel passwords, contact Robelle for assistance.

- **Initial Directory**

This text box defines the non-home directory that Qedit will open when you start the program. The directory you specify in this box will continue to be associated with your user name until you change it. If you leave this text box blank, Qedit will put you in your home directory when you start the program.

Although the user ID is used to log on, Qedit changes to the initial directory as soon as the connection is established by performing a *cd* command relative to the *root* directory, not to the user's home directory. For example, if you want to automatically go to `/users/demo/public_html`, you can enter

```
/users/demo/public_html (explicit)
```

or

```
users/demo/public_html (implicit)
```

Even if `/users/demo` is the home directory for the demo user, you *cannot* simply enter `public_html` or `./public_html` because you will get this error:

```
Login: Unable to change directory.
```

Copying a Saved Connection

You can use an existing connection as a template for a new connection. To copy a connection,

1. Select the **Connections** command in the **Options** menu.
2. When the **Connection List** dialog box appears, select the connection you want to copy from the **Connection Name** list.
3. Give the connection a new name. This will copy the original connection and create a new connection.

The original connection does not change; it remains in the connection list. This allows you to use an existing connection as a template for new ones.

NOTE: For the time being, we recommend that you retype the passwords on new connections.

4. If you have modified only the host login or passwords, click **Update** to save the new values.

If you have created a new connection, click **Add**.

5. Click **Done** to close the dialog box.

Opening a Connection

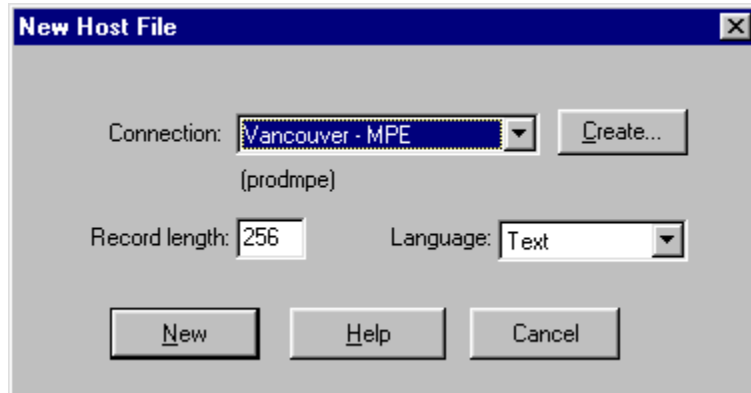
Opening a Temporary Connection

As soon as you define a temporary connection, you can open it. A temporary connection stays open until you quit Qedit for Windows.

Opening a Saved Connection

Once you have added a saved connection to your list, you can use it repeatedly and access it quickly. To open a saved connection,

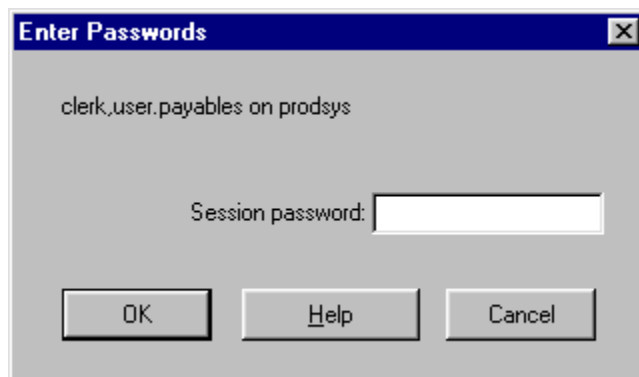
1. Select the **File** menu, point to **New**, and click **Host**. You will see this dialog box.



New Host File dialog box

2. Select a connection from the **Connection** list, which is sorted alphabetically. That means MPE and UNIX connections are interspersed.
If you want to create new connections once you start opening a file, use the **Create** button.
3. Click the **New** button to open a new empty scratch file on the designated connection.
4. If you have configured this connection to prompt for a password, Qedit displays a dialog box with the **Login** or **Hello** string and the **Host** name as well as the number of text boxes required to enter all the passwords you requested.

Enter all the passwords and click **OK**.



Password prompting dialog box

Once you have entered the passwords, you will see a series of progress messages followed by a new document window labeled **Untitled 1**.

If there are problems with the connection, first check your spelling and punctuation. It is also possible that the problem is at the host end: the wrong version of Qedit, networking not enabled, backup in progress, etc. See the section on troubleshooting

“MPE Host Connections” on page 211 or troubleshooting “HP-UX Host Connections” on page 216.

Editing a Saved Connection

A connection is defined as a connection name, a host name, a user login name for UNIX or a “hello” string for MPE, and the passwords needed to access that host. (Qedit for Windows does not bypass any of your security provisions.) You can change the host name, the login information, the passwords and the options in a saved connection.

If you change the connection name, you are really creating a new connection. The old connection is not renamed and it remains in the connection list. This allows you to use an existing connection as a template for new ones. If you want to change the name of an existing connection, you need to create a new one and delete the old one.

Modifying a Saved Connection

To modify a saved connection,

1. Select the **Connections** command in the **Options** menu.
2. When the **Connection List** dialog box appears, select the connection you want to change from the **Connection Name** list.
3. You can change the host name, the login information, the passwords and its prompt options on a connection. You cannot, however, change the connection name because that would actually create a new connection. See “Copying a Saved Connection” on page 56.
4. Click **Update** to save the new values.
5. Click **Done** to close the dialog box.

Any subsequent use of this connection will implement your changes.

Renaming a Saved Connection

If you want to rename a connection, it is a two-part process: first you need to rename the original connection when you make a copy of it, then you need to delete the original connection.

To copy the original connection to a new connection name, see “Copying a Saved Connection” on page 56. To delete the original connection see “Deleting a Saved Connection” below. When you have finished these steps, the connection list should not include the original connection.

Deleting a Saved Connection

If you no longer need a saved connection in your list, you can delete it. To delete a connection,

1. Select the **Connections** command in the **Options** menu.
2. When the **Connection List** dialog box appears, select the connection you want to delete from the **Connection Name** list.
3. Click **Remove**. The connection should no longer appear in the connection list.

4. Click **Done** to close the dialog box.

Closing a Connection

Unless you change its default, Qedit does not automatically close a connection when it has no open files. To close a connection, select the **Disconnect** command from the **File** menu. If you have more than one connection, select the connection you want to close.

To ensure that all of your connections are disconnected, select **Exit** from the **File** menu. This closes all files and disconnects all hosts.

Customizing a Connection

Automatically Logging In at Startup

By default, Qedit does not set up a connection or log in to a host at startup. If you want Qedit to automatically log in at startup, you can change your default.

1. Select the **Connections** command in the **Options** menu.
2. Select the connection to which you want Qedit to log in at startup.
3. Select **Login at startup**. The next time you start Qedit for Windows, it will create this connection for you automatically, but it does not open any files. You can enable this option when you first create a connection.

Closing Connections without Open Files

By default, Qedit keeps connections open until you disconnect them implicitly or until you exit from Qedit. Sometimes you might not want to keep a host connection open if it is not being used. For example, if the likelihood of ever reusing the connection is slim, you might not want to maintain the resources.

You could use this option if you are worried about people accessing the host through your PC and you are willing to close all files when you walk away. A screen saver with password protection, however, might be a better solution for this problem.

To change Qedit's default so that it automatically closes connections without open files,

1. Select **Preferences** from the **Options** menu.
2. When the Qedit Preferences dialog box appears, select the Close connections with no open files option.
3. Click the **OK** button to close the dialog box.

This option forces Qedit to close not only connections, but also open directory dialog boxes on connections that are forced to close.

Creating & Opening Files

Creating Files

You can create a new file on an MPE or HP-UX host or on your PC. In Windows terminology, creating a file from scratch is called the **New** operation, while **Open** usually refers to an existing file.

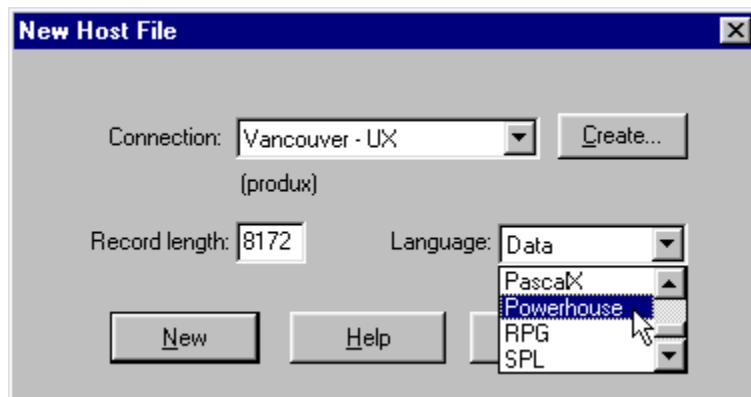
Creating a New Host File

Shortcut: CTRL+SHIFT+N

Toolbar: 

To create a new file on a host system,

1. Select the **File** menu, point to **Host**, and click **New**.
2. When the following dialog box appears, define the file's characteristics:



New Host File dialog box

- First select a **Connection**, which is a host system and specific login name and security. If you need to use a connection repeatedly, you can create a new saved connection. (See “Creating a Saved Connection” on page 52). Once you have created a saved connection, it is accessible through your connection list until you remove it from the list.
- Next, select a **Language** for your file. This is not a spoken language such as French, German, or English, but a computer language like COBOL, C++, PowerHouse, or Data (the default). The **Language** is important because it determines how the **Smart string** option will work (i.e., hyphen

is valid in a word in COBOL, but not in Pascal), and it defines the maximum **Record length** for your file.

Qedit for HP e3000 is able to recognize the **Language** setting and select the appropriate compiler based on that. At the moment, you need to run host-based Qedit from a terminal or terminal emulator to take advantage of this feature.

Many PC editors set the maximum line length to 32,000 or 64,000 characters, so the concept of **Record length** doesn't really apply. Qedit is designed for editing host files, which often have a maximum length for each line. For example, COBOL has a fixed format in which the first 6 columns contain a sequence number, the next 66 contain the program code, and the last eight are comments. By default, Qedit opens a Data type file with a **Record length** of 256 characters, but you can increase or decrease that value. Qedit's maximum record length is currently 8,172 characters for a host file.

3. Click the **New** button. Qedit will create a connection to that host (if you do not have one already) and attempt to create a new file in your login directory. Until this file is given a permanent name, you will see the connection name followed by **Untitled1** in the window title bar.

Once you have entered some information into your new file, use the **Save** command of the **File** menu to save the file. This automatically gives you the **Host Save As** dialog box in which you can assign a new name to the file. Otherwise, the file will be lost if the connection terminates for any reason.

Creating a New Local File

Creating a new local file on your PC is very similar to creating a new host file. Just select the **Local** submenu on the **New** command in the **File** menu. Qedit for Windows creates a new document window on your PC with these defaults: Data as the **Language** and a maximum **Record length** of 8,172 characters.

Creating Local Directories

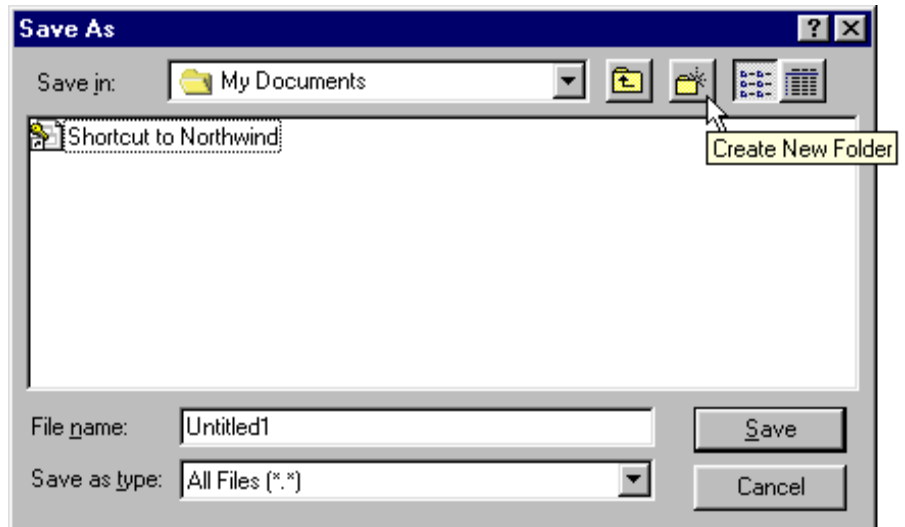
The method for creating a directory on your PC depends on your version of Windows. For Windows 95 and Windows NT 4.0, follow these steps to create a new directory for a newly created file:

1. On the **File** menu, point to **New**, and then click **Local**.
2. Type the information in the file.
3. On the **File** menu, select the **Save** command.
4. When the **Save As** dialog box appears, click **Create New Folder** (its icon looks like a folder with a star). A new folder will be created within the current folder, which is the directory in the **Save in** text box. The folder is assigned a default name such as **New Folder** and is highlighted so that you can rename it to something meaningful.

This option is also available on other Windows dialog boxes.

Shortcut: CTRL+N

Toolbar: 



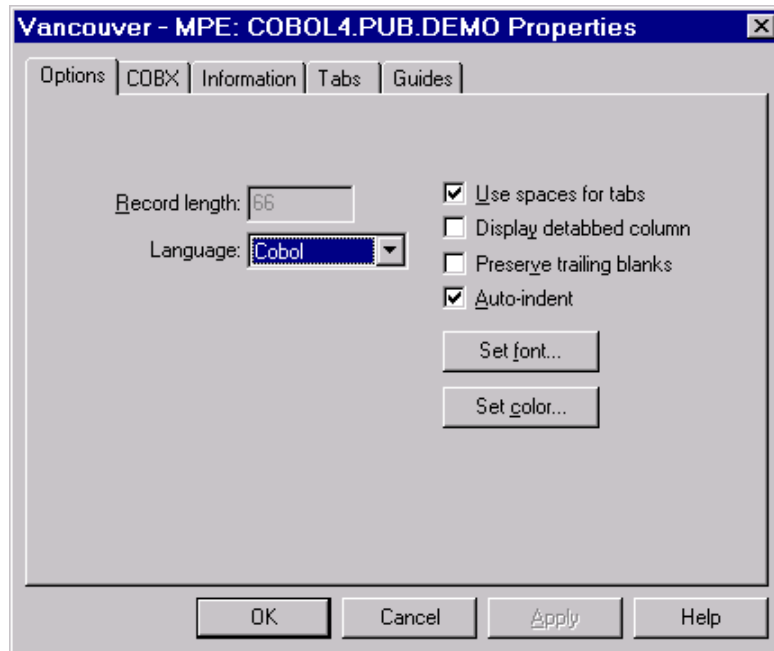
Creating a new folder using Windows 95 or Windows NT 4.0

Defining File Options

Each file has its own set of file characteristics. These characteristics are saved in the document database. New files, files that are not in the document database i.e. files that have never been opened and files in the document database that do not have customized file properties, acquire the default settings defined in the **Preferences** dialog box. Once the file is in the document database, Qedit uses the saved characteristics on subsequent access. To check or change the characteristics of a file,

Shortcut: ALT+F,I

1. Select the **File** menu and point to **Properties**.
2. When the **Properties** dialog box appears, you will see the following options:



File Properties dialog box

- **Record length**

The **Record length** text box indicates the number of columns available on each line. The value shown in the example above is the default for a COBOL file from an HP e3000 host. It has a maximum of 66 columns per line, because columns 1 to 6 are used for a sequence number and columns 73 to 80 are reserved for comments. If you want to include comments in COBOL, select **COBX** in the **Language** list.

Local files can have a length between one and 8,172 characters per line.

- **Language**

The **Language** list allows you to choose from a number of different programming languages and their file types. Each language has different characteristics with the **Record length** being the most evident.

The **Language** setting also affects the way Qedit searches with the **Smart Search** option and what gets selected when you double-click a word. Qedit supports the host-defined rules for searching.

- **Set Font**

To access the font settings, click **Set Font**. When the **Font** dialog box appears, select the **Font**, the **Font style**, and the **Size** from the list boxes. If you want a font size that is *not* available in the list, you can type a custom size in the text box.

- **Set Color**

To change the color settings, click **Set Color**. When the **Color customization** dialog box appears, select the color scheme and, if you like, change individual color elements. For detailed instruction, see "Setting the Default Color Scheme" on page 121.

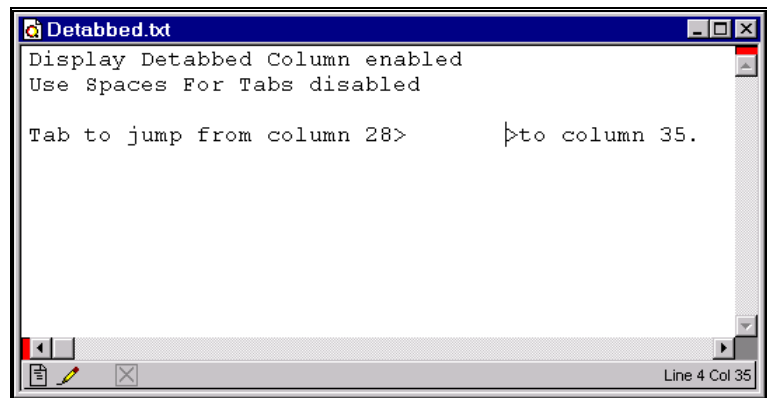
- **Use spaces for tabs**

When enabled, Qedit for Windows replaces horizontal tab characters in the file with the appropriate number of spaces. For HP e3000 host files, the **Use spaces for tabs** option is enabled as the default. For HP-UX hosts and local PC files, this option is disabled as the default.

- **Display detabbed column**

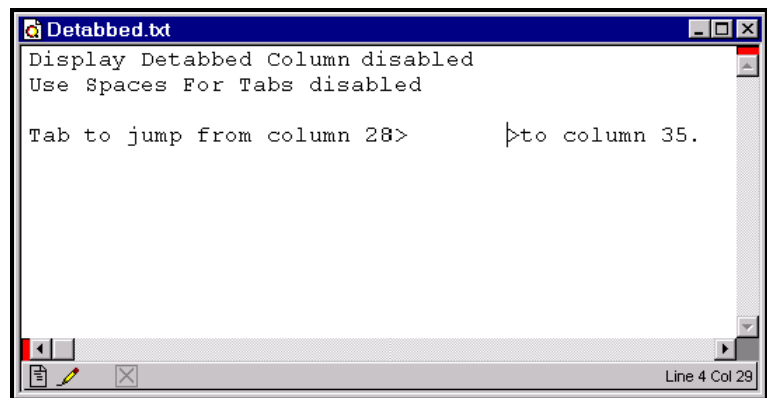
The status bar at the bottom of a document window displays the cursor position relative to the beginning of the file. The status bar shows the relative line number and the column number within the line. If the file contains tabs and you press the TAB key, Qedit moves the cursor to the next defined tab stop and adjusts the column number accordingly. If there is text to the right of the cursor when you press TAB, the text will also be moved.

With **Display detabbed column** enabled, the column number on the status bar includes the number of spaces the TAB occupies, as if the TAB character had been replaced with spaces (see **Use spaces for tabs** above).



Document window with Display detabbed column option enabled

If **Display detabbed column** is disabled, the column number on the status bar counts the TAB as one character, regardless of the number of spaces it occupies. In the example below, the text following the TAB is displayed in column 35, but the column number only shows a value of 29.



Document window with Display detabbed column option disabled

- **Preserve trailing blanks**

By default, Qedit preserves trailing spaces that have been entered. If you try to insert new text on these lines, the trailing spaces will eventually reach the right margin and you won't be able to insert any more characters. When this happens, you have to go to the end of the line and remove some spaces to make room for the extra character(s). Although this process is awkward, Qedit uses it as a default because it presumes that anything you enter is valuable and should be kept intact.

Unfortunately, you cannot tell whether or not a line has trailing spaces. If trailing spaces are really not important, you can disable this feature. Once disabled, Qedit removes the necessary trailing spaces as you enter new text. Note that if you delete characters from a line, Qedit does not return the trailing spaces that were removed.

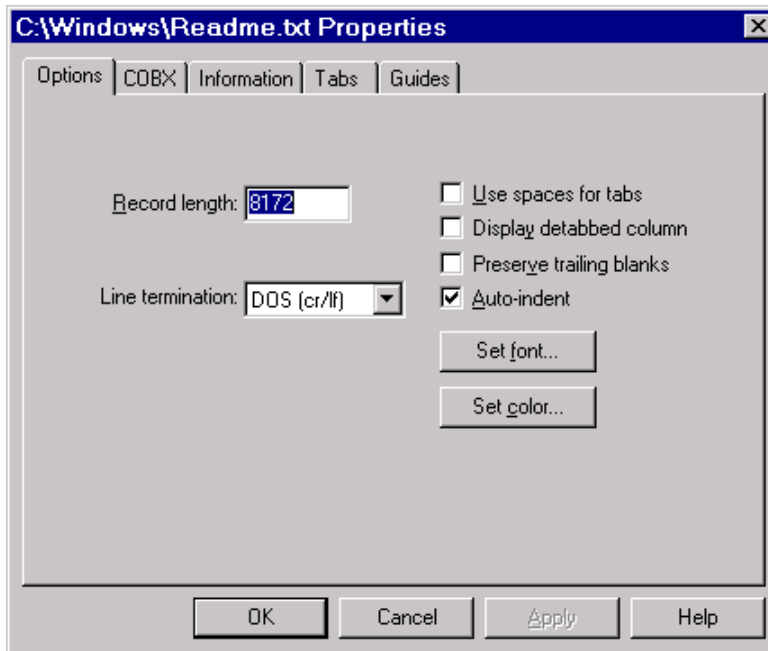
This setting only affects editing while the file is opened. As you modify the file and add trailing blanks, Qedit recognizes them. However, trailing blanks are not written to the file when you save it. Next time you open the file, trailing blanks are gone. That's because the server never transmits them to the Qedit client.

- **Auto-indent**

Auto-indent is a great feature because it allows you to follow the indent of the last line when you press the RETURN or ENTER key. Instead of moving the cursor to the first position in the line, it goes to the column that matches the indentation of the previous line. You can use spaces or tabs to increase the indentation. If you use tabs, you can use SHIFT+TAB to decrease the indentation.

Setting the Line Termination Character

These file properties discussed above are common to all types of files. Local files have an additional option that controls which line termination character is used when a file is saved to disk. Qedit automatically detects which line termination was used originally and, by default, uses the same one when the file is saved. You need to change the Line termination character only when you want to explicitly change the file type.



Local file Properties dialog box

The **Line termination** list box offers the following selections:

- DOS (CR/LF)
- Macintosh (CR)
- UNIX (LF)

CR stands for carriage return and LF for line feed. This option only affects the way files are created on save operations. It is a useful feature when you know the file is going to be used on a different platform and want to ensure the file is created appropriately for the target application.

Notation in Host Operating Systems

Qedit for Windows can work on hosts with MPE, POSIX and HP-UX. To move through directories in each of these operating systems, you must be familiar with their file structures and notation conventions.

UNIX Notation

The directory structure in UNIX is hierarchical. The slash (/) at the beginning of a path name represents the root directory. There are two ways to specify a path name: absolute or relative. An absolute path name is the path relative to the root directory. A relative path name is a path relative to your current working directory.

Each file name or directory name follows its parent directory and is separated from it by a slash (/). UNIX names, which can be in either case, are case-sensitive. These are some examples of valid path names:

```
test/script1
/users/bev/chk.txt
../src/abc.c.
```

MPE Notation

Names in the MPE namespace are always shown as uppercase. MPE file names are not case-sensitive (e.g., `p@` is the same as `P@`). An MPE file called `Cobol4` can be entered in either case. These are examples of MPE files:

```
file
file.group
file.group.account
```

To access a file by its hierarchical name on MPE, you must precede the name by a dot-dot (`..`) to go to the parent directory or one level up in the hierarchy, use dot-slash (`./`) to go to the current directory, or use slash (`/`) to start at the root directory.

```
../group/file
./file
/Account/Group/File
```

POSIX Notation on MPE

With MPE/iX 4.5, HP introduced POSIX support. POSIX is an acronym for Portable Operating System Interface. Because the POSIX standard is based on BSD UNIX and AT&T's System V Interface, MPE has been extended to support an Hierarchical File System (HFS) similar to the one found on UNIX hosts. As a result, UNIX file name syntax (with some restrictions) can be used to navigate the POSIX namespace.

Important Note: Some restrictions apply if your MPE host is running *version 4.0* of the operating system. You can use the **Directory** dialog box to view and select files only in the logon group. Qedit uses a POSIX feature of MPE to move between groups and directories. Because POSIX is only available with MPE/iX 4.5 or later, you will not be able to switch to a different group dynamically. If you need to use the directory listing of another group, you have to create a separate connection that logs on immediately to the appropriate group.

MPE files are also accessible using this notation. Whenever possible, Qedit displays names using MPE notation. However, if you move to the POSIX namespace, Qedit adjusts the current working directory to POSIX notation. MPE names shown in absolute POSIX notation have the account name first followed by the group name and finally the file name. For example, the `Cobol4` file is accessed via POSIX as `/Green/Pub/Cobol4`.

Names in the POSIX namespace can be in either case, but they are case-sensitive. POSIX directories end with a slash character (`/`).

POSIX names are identified by a leading SLASH character (absolute) or a leading dot-slash or dot-dot-slash (relative). If you don't specify one of these prefixes, Qedit assumes you are trying to access a file in the MPE namespace. If you want to open a file using POSIX notation, you could enter `/GREEN/PUB/COBOL4` or `./COBOL4`.

You have to use MPE wildcards when you access the POSIX namespace. If you want to select all the files that start with a capital `p` on an MPE host using the POSIX notation, you would use the at sign (`@`) as the wildcard. So, you could enter `./P@`.

Host Directories

When you want to open a single file and you know its name as well as the directory in which it resides, you can use the **Open** command. See "Opening a Host File" on page 77.

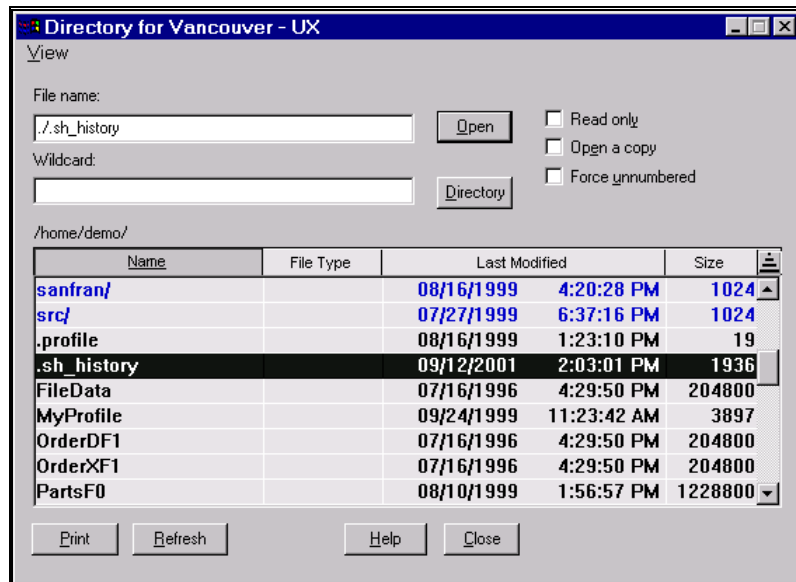
Going into the file structure of a host, however, is useful in the following situations:

- you don't know the exact name of a file
- you don't know the directory in which a file resides
- to open multiple files at once
- to browse the contents of a directory

Accessing Host Directories

To access a host directory,

1. On the **File** menu, point to **Open**, and then click **Host**.
2. When the **Open Host File** dialog box appears, click **Directory**.
3. When the **Directory** dialog box appears, it will list the files and subdirectories in the current working directory or group. This dialog box also includes text boxes (**File name** and **Wildcard**) for searching a file and buttons to perform further operations.



The Directory dialog box

The Directory Listing

The first entry in a directory listing is always two dots (..), which is the parent directory of your current working directory. Directories (parent or subdirectories) within the listing appear in color. The listing includes all files, even those starting with a dot; these files are usually hidden in UNIX.

NOTE: Currently, only MPE displays File Type information.

The directory listing appears in alphabetical order. The listing includes the following information: the file **Name**, the **File Type**, the date and time **Last Modified**, and the **Size** of the file. Although the **File Type** column appears on both MPE and UNIX systems, only MPE currently includes this information.

The **Last Modified** date and time is based on the local time on your PC. Qedit assumes the clocks on all your computers are correct and takes care of adjusting the time across different time zones. See “Time Zones” on page 221.

The **Size** column represents the number of records for MPE files and the number of bytes for UNIX files.

If you want to make sure you have the latest information in the directory listing, click the **Refresh** button. This forces the server to discard the current list and bring in the latest set of files. Note that the wildcard selection is ignored on a **Refresh** operation (i.e., Qedit displays a complete list of files in the current working directory or group).

Sorting the Directory Listing

Order Selection button: 

By default, Qedit sorts files in ascending order. You can reverse the sort order by clicking the **Order Selection** button located on the right side of the heading line. When the arrow on this button points up, the sort order is ascending; when the arrow points down, the order is descending. To change the default sort order, select the **Sort order** command in the **View** menu.

You can sort the list on any of the headings (**Name**, **File Type**, **Last Modified** and **Size**) by clicking the column heading. The heading for the currently selected sort item appears on a darker background, as if the button was pressed down.

Searching for a File

These are some of the ways you can search for files on host directories:

- Scroll through the current directory. You can use your mouse or cursor movement keys to move through the list of files and subdirectories. The highlighted file indicates your current position within the list.
- You can go to other directories. See "Changing Directories" on page 71.
- If you know part of the file name, use the **Wildcard** text box to search through the current directory. See "Using Wildcards" on page 70.
- For a more precise search, you can combine the **File name** and **Wildcard** text boxes. If you want to search another directory for example, enter its path name in the **File name** and use the **Wildcard** text box to narrow your search.

Using Wildcards

Wildcards are symbols that represent unknown characters. If you know part of the file name, use the **Wildcard** text box to find all files that match. Unless the **File name** text box contains a different directory, Qedit will search in the current working directory.

Make sure you use the host-appropriate rules for wildcard characters because these rules are different for MPE and UNIX. Qedit does not do any validation on the string.

On UNIX for example, an asterisk (*) means one or more characters, and file names are case-sensitive. So if you want to see all files that start with a capital *p*, you would enter P*. Please refer to the appropriate host user manuals for details on using wildcards.

The Listfile command with Seleq parameter is not available on all versions of MPE.

Qedit passes whatever is in the **Wildcards** text box to MPE's Listfile command. If you specify a wildcard that does not start with an HFS prefix (slash [/], dot-slash [./] or dot-dot-slash [../]), Listfile assumes the information is for an object (file, group or account) in the MPE namespace. If you use one of the HFS prefixes, Listfile searches the POSIX namespace. Currently, there is no way to get a complete file list by searching both namespaces with a single wildcard string.

Under MPE/iX 5.5 (and perhaps other versions), you can use the Seleq parameter to further refine the requested file set.

For example, you could use the Code= option of the Seleq parameter to only select files of a certain type. You would then be able to select only Qedit files (code 111), CM program files (code PROG), NM program files (code NMPRG) or COBOL source files (code EDTCT) using wildcards such as:

```
@;seleq=[code=111]
@;seleq=[code=prog]
@;seleq=[code=nmprg]
@;seleq=[code=edtct]
```

For a list of file types, refer to "Appendix B - File Types" on page 231. The Seleq parameter has a lot of different options. Please refer to your host user manuals for details.

Changing Directories

Qedit displays directories (parent or subdirectory) in color. If you select a directory within the current working directory, it becomes the new current working directory. You can look for a file by continuing to move to other directories in which you have read access.

It is important to remember that there is only one current working directory per connection. Even though you can go to the **Directory** dialog box from the **Open Host File** and the **Host Save As** dialog boxes, you are really dealing with the same **Directory** dialog box. This means a change in the current working directory reflects on both host Open and Save As operations.

If you know the exact location of a directory, you don't have to go through other directories to get to it. Just use the absolute (e.g., /users/demo) or relative (e.g., ../demo) syntax of the path name in the **File name** text box and select the **Directory** button.

For more host-specific information about file and directory notation conventions, see the "Notation in Host Operating Systems" on page 67.

Selecting Files

Once you have found the files you need, they can be opened from the **Directory** dialog box. You can select a file by highlighting it in the listing (it then appears in reverse video) or by entering its name in the **File name** text box. To highlight a file, click it once; to remove the highlight, click it again.

If you want to open a number of files at once, you can highlight them in the listing. You can select up to ten (10) files. When you select multiple files, the **File name** text box will display the number of files selected. You cannot use the SHIFT key to select additional files; you have to select them one by one.

Opening Selected Files

Qedit allows you to open files from the **Directory** dialog box. Once you have selected a file or a number of files, simply click the appropriate button.

When opening files on a UNIX host or in the POSIX namespace on MPE, Qedit prepends the file name with dot-slash (./) to indicate the current working directory, if the file name does not start with a dot (.) or a slash (/) that already indicate a relative or absolute path.

If you have an MPE file and a POSIX file that have similar names, the server might not open the file you expect. For example, let's say the MPE file is called `MYFILE` and the POSIX file is called `myfile`. Qedit does not prefix the POSIX file with `./` because it does not distinguish lowercase from uppercase letters. Aside from the case, the names are identical. In this situation, the server always finds files in the MPE namespace first. Thus, if the filename is not explicitly identified with `./`, `MYFILE` will always be opened.

Printing a Directory Listing

You can get a copy of information displayed in the **Directory** dialog. Simply click the **Print** button. The listing can only be printed to a local printer.

Saving Host Files

If you have invoked the **Directory** dialog from the **Host Save As** dialog, the **Open** button becomes **Save**. You can navigate through the directory structure until you get to the desired location. You can enter a new name or you can select one from the list. To save the file, click **Save**.

Opening a File

Very often, you will be working on existing files. In order to modify these files, you first need to open them. Qedit can open files in a number of ways, including some shortcut methods. Qedit always applies a file's saved attributes from its document database, no matter how the file is opened.

Document Attributes Saved by Qedit

Qedit maintains a document database in which it stores document-specific attributes. Document information is added or updated whenever files are opened, saved or closed.

Local file names are fully qualified including the drive letter. Host file names are also fully qualified including the connection name on which they were opened.

For each file in the database, Qedit keeps track of the following settings:

- the cursor position or selection active at the time of closing
- the display status of the **Ruler bar** and **Record numbers**
- current **Tab** settings
- current **Guide** settings

- all of the check-box settings from the **Options** page of the **Properties** dialog box
- the font settings from the **Options** page of the **Properties** dialog box
- the document-specific COBOL tag

Qedit is careful to track only these 6 settings for each document. They are stored locally in the database called Qwinddb.dat. See “Support Files” on page 11. Nothing is stored on the server. That means you can copy the document database from one PC to another and get the same attributes on all client machines (e.g., your home PC and your PC at the office). To have complete consistency between machines, you also have to make sure certain information, such as the connection names and available fonts, is the same on each machine.

Qedit saves the last selection position within the file as line and column coordinates in relation to the beginning of the file. If a file is modified with something other than Qedit, next time you open the file with Qedit, it selects the same coordinates, but the selected text might be a completely different string.

If you open a host file and view the ruler, you will see the default tab settings specified in the host Qeditmgr files. If you close the file, exit Qedit, start it up again, and open the same file, you will continue to see the default host tab settings, even if these settings changed on the host.

If you open a document, view the ruler, and then change the tab settings, these become the new tab settings for the document. They override whatever default settings were specified in the host configuration files.

Similar rules apply to the font settings. If you never change the font of a document when you open it, you will view the document in the default font. If you do change the font of a document, the new font will be saved and it will be applied from that point onward. It does not affect other documents.

There is currently no way to reset all the local and host defaults for a modified document.

Opening a File from a Command Line

The command line interface feature in Qedit can automatically open a file at startup. You can use this feature from a DOS prompt, in a DOS shell script, or in any other application that has a command line interface. You cannot create a new file from a command line; you can open only an existing file.

The command line feature uses the following syntax:

```
qedit <connection>:<filename>
```

Qedit is the program file name. In a **Typical** installation (on a stand-alone PC), the program file name would be Qwin32. Depending on the interface you use, you might have to qualify it with the letter of the drive, the full path name (e.g., C:\Robelle\Qwin32) and the file extension (e.g., Qwin32.exe).

Connection represents the valid name of a saved connection, which must be identical to a name in Qedit’s connection list. This means you must make sure all lowercase and uppercase characters are entered correctly, as well as any spaces and special characters.

Qedit assumes everything in front of the colon is part of the connection name. If it cannot find a corresponding name in the connection list, it assumes the file is local

and tries to open it as such. To open a local file, you can simply use the letter of the drive as the connection name. For example, you can enter

```
qwin32 c:\windows\readme.txt
```

Everything after the colon is read as the file name. You can use relative or absolute notation, just the file name, or a fully qualified file name, but you must use the notation appropriate for the operating system on which the file resides.

Qedit tries to open the application window with the "restore" size from the last edit session. If the "restore" size is not large enough, Qedit will not start properly and display an error. To avoid this problem, you can use the "-m" argument to force it to start with a maximized window.

Opening Embedded Files

The fastest way to open a file which has its name embedded in your document (e.g., an Include file), is by highlighting it and entering CTRL+D. Or you can open it by right-clicking on the highlighted selection to get the shortcut menu, and choosing the **Open** command. You can also open it by selecting of the **File** menu, pointing to **Open**, and clicking the **Open** command with the selected file name.

If the host file you are opening is not fully qualified, Qedit will look for the file in the current working directory (UNIX) or the current group (MPE) on the same connection as the current document.

If you want to open a file on a different connection, you can specify the connection name (as defined in the connection list) as a prefix to the file name, separated by a colon. For example,

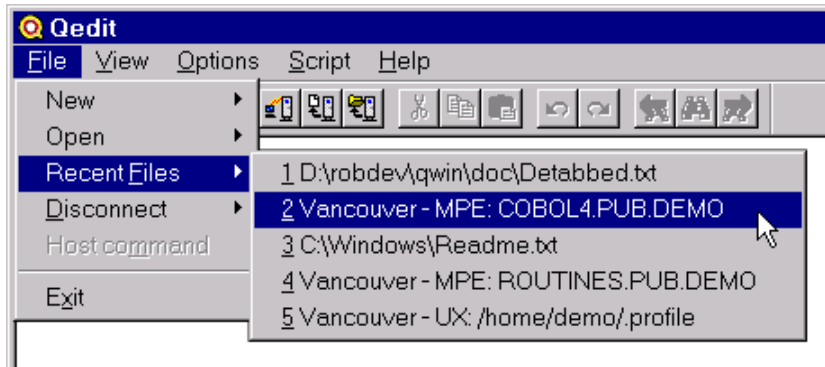
```
Vancouver MPE:myfile.mygrp  
Vancouver UX:/users/clerk/.profile
```

are valid file names. If you select one of these strings and use the **Open** selection command, Qedit will verify that Vancouver MPE or Vancouver UX is a valid connection name. If so, it will automatically try to connect to the corresponding host and open the file Myfile.Mygrp or /users/clerk/.profile.

Opening MRU (Most Recently Used) Files

Qedit maintains a list of the most recently used files. This list is accessible via the **Recent files** command in the **File** menu. Local file names are fully qualified including the letter of the drive. Host file names are also fully qualified including the connection name on which they were opened. For host files, Qedit also keeps track of the options used the last time you opened the file (i.e., **Read-only** or **Force unnumbered**).

To open a file in the list, select the **Recent files** command in the **File** menu. Qedit presents the list of most recently used files as a submenu.



List of most recently accessed files

By default, Qedit for Windows displays 20 files. This number can be changed in the **General** tab of the **Preferences** dialog. Simply enter the desired number of files in the **Length of Recent Files or Favorites lists**. You can enter a value between 5 and 50.


When the filelists are displayed, a shortcut character is assigned to the first 35 files. This means you can select a file by combining the **ALT** key with corresponding character as in **ALT+3** or **ALT+B**. The first are numbered from 1 to 9. The next 26 are labeled from A to Z. Files 36 through 50 do not have a shortcut character.

When you open a file, Qedit always applies the saved document-specific attributes from its document database, no matter how the file is opened. You do not have to use the **Recent files** command to take advantage of this feature.

Opening a Local File

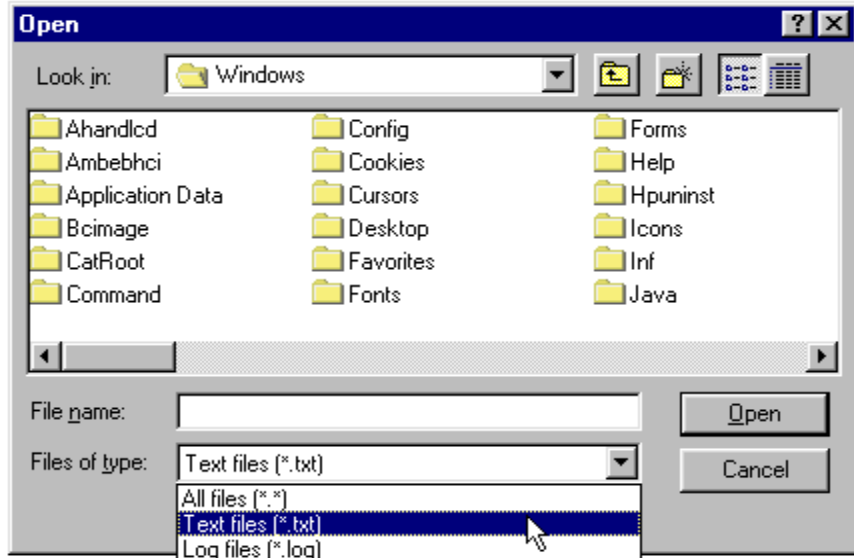
Qedit can open local files, including text and HTML files. To open a local file on your PC or on a networked drive,

Shortcut: CTRL+O

Toolbar: 

1. Select the **File** menu, point to **Local**, and click **Open**.
2. When the **Open** dialog box appears, it provides the following options:
 - **File name**
 - **Files of type**

By default, Qedit displays all files and folders available in the current directory. You can reduce the number of files by selecting one of the filters available in the **Files of type** list box. The default is **All files (*.*)**. Click the list box to display available filters and select one. Note that the filters do not apply to folders in the directory. Only files are filtered.



Select a file filter on local files

- **Open as read-only**

If you open a local file with write access (the default), you can, of course, make changes to it. If you save these changes, you effectively erase the old version of the file. You can, however, protect yourself from modifying a file by enabling the **Open as read-only** option before you open the file.

Windows will mark a file opened with write access as busy. In Windows 95, some applications automatically convert any subsequent opens into read-only mode. Qedit allows you to open the file again, but it will give you an “Error deleting a local file—permission denied.” error message when you attempt to save changes to the file. Qedit does not stop you from opening the file again.

Before you open a file, the buttons and the directory listing on the **Open** dialog box allow you to perform the following operations:

- navigate through folders
 - create a new folder (**Create New Folder** button)
 - list the contents of a directory (**List** button)
 - list the details of a directory (**Details** button)
 - display all files of a specific type
3. Open a file by double-clicking it or by entering a **File name** and clicking the **Open** button.

You can also select multiple files and open them all at once. Qedit uses Windows’ standard way to select multiple items.

There is a limit to the number of files that can be opened in a single operation. The limit is not the number of files, but rather the number of characters that make up all these file names plus the drive letter, the current working directory and name delimiters. All these cannot add up to more than 2,048 characters. If you exceed this limit, Qedit displays an error message after you press the **Open** button.

Opening an HTML File

In recent versions of Microsoft Internet Explorer, you can edit the Web document that you are currently viewing. By default, when Internet Explorer opens an HTML file, it runs Notepad or Wordpad depending on the size of the document. To change the default editor for an HTML file to Qedit,

1. Start Windows Explorer (not Internet Explorer).
2. Select the **Options** command in the **View** menu.
3. Select the **File Types** tab.
4. Find Internet Document (HTML) or Microsoft HTML Document 4.0 in the **Registered file types** list. The actual wording can be a little bit different. Just make sure the file extensions are HTM and HTML.
5. Select **Edit** in the **Actions** list. If it is not there, select **New**.
6. In the **Application used to perform action** text box, enter the fully qualified name for the Qedit program. You can select **Browse** to find it using Windows Explorer.



Internet Explorer Edit tool

You can edit a Web document by selecting the **Page** (Internet Explorer 4.0) command in the **Edit** menu or selecting the **Edit** tool. A new instance of Qedit will run automatically with the current document opened. You are then able to make changes to the document.

You can save your changes to local files without any problems. However, if the Web document is on a remote server (i.e., an MPE or UNIX host), Internet Explorer simply passes a temporary copy of the document to Qedit. In this case, the only thing you can do is to use the **Save As** command of the **File** menu and save the revised version under a new name on your local drive. You then need to transfer the file back to the server (e.g., using FTP).

Opening a Host File

Qedit can open MPE and HP-UX host files, Qedit workfiles as well as copies of these files.

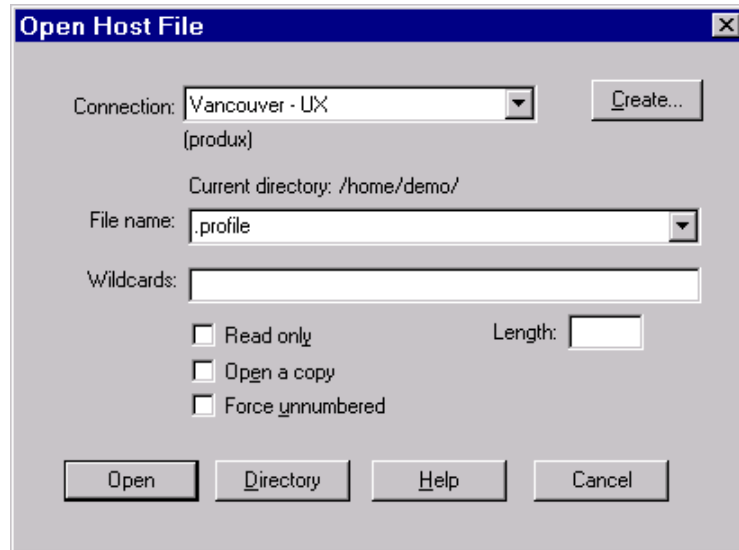
WARNING: Opening binary files with Qedit can corrupt them. See “Opening Files with Binary Data” on page 80.

To open an existing file on a host server,

Shortcut: CTRL+SHIFT+O

Toolbar: 

1. Select the **File** menu, point to **Host**, and then click **Open**.
2. When the **Open Host File** dialog box appears, it provides the following options to either search for a file or to open it:



Open Host File dialog box

- **Connection**

Select a connection from the **Connection** list. If you need a new connection to a host or you need to use a new login name, click the **Create** button. You can also use the **Create** button to set up a temporary connection.

If a host file is the active window when you access this dialog box, Qedit opens the file on the same connection as the active file. If no document is open or if the active window is a local file, Qedit uses the connection name that was used to open the last host file.

- **File name**

Enter the file name or choose a file from the most recently used (MRU) list. This list includes files that have not been opened successfully, starting with the most recent name at the top of the list. The list is preserved from one Qedit session to the next. The file names are recorded as you originally entered them, even misspelled files. Qedit does not keep track of the connection associated with a file name in the list.

If the file you want is not in your home directory on HP-UX or logon group on MPE, you can type the path name in the **File name** text box.

NOTE: Files opened via the **Directory** dialog box, the **Open selection** command (CTRL+D) of the **File** menu, or a shortcut menu (right-click) are not added to the MRU list. The list also does not include files saved by the **Save as** command. However, *all typed file names* are added to the MRU list, which is available through the **Recent files** command in the **File** menu.

- **Wildcards**

Because the **File name** and **Wildcards** text boxes are also in the **Directory** dialog box, read “Host Directories” on page 68 for general information. For more host-specific details, read “Notation in Host Operating Systems” on page 67 for the corresponding host: UNIX, MPE with POSIX and MPE. These sections describe how to use the text boxes

along with the **Directory** button to quickly switch to other directories and groups, and to select only the files that you want.

- **Directory**

If you don't want to type or you can't remember the spelling of the directory or file, there is another way to find a file. Just click the **Directory** button. This will give you a list of files in the current directory and access to other directories on the host. For more details, see "Host Directories" on page 68.

If you find the file through the **Directory** dialog box, you can open it from there.

- **Read only**

Sometimes you know that you just want to look at a file, but you don't want to modify it. In this case it is better to enable **Read only** (ALT+Y) before you open it, which will protect you from making any accidental changes. This option is also useful if someone else is editing the file you want to view. The **Read only** option is available in the **Open Host File** and **Directory** dialog boxes. If you have selected multiple files in the **Directory** dialog box, the **Read Only** option will apply to all of them.

- **Open a copy**

There are times when you want to modify a file, but you don't want to lose the last version of it. To open a copy of a host file, enable the **Open a copy** (ALT+E) option before opening the file. For more ways to copy a host file, see "Copying a Host File to Another Host" on page 149.

- **Force unnumbered**

The server analyzes the last 8 columns of the first 5 lines in a file. If these columns contain only numeric digits and these digits are in ascending order, Qedit assumes these are line numbers and should not be part of the data. In certain cases, this can cause problems because these numbers might very well be valid data.

To force Qedit to read these files as unnumbered, enable the **Force unnumbered** option (ALT+U) before opening the file. The numbers will be displayed along with the rest of the text. You will also be able to edit these columns.

- **Length**

This option is available on HP-UX connections only and with version 5.3.13 or later of the server. If the connection has not been opened yet, the option is available but might be ignored by the server. If the connection is opened and the server does not support the feature, the option is not displayed at all.

If the file does not contain Newline delimiters at the end of each line, you can enter a maximum line length to break the file into pieces. In this case, all lines will have the same length except for the last one which might be shorter. This would happen if the total size of the file is not evenly divisible by the specified length.

You can specify a maximum line length on a file that contains Newline delimiters. In this case, lines that are shorter than the specified length will remain the same. Lines that are longer than the specified length will be

broken up in pieces. If the existing line is not evenly divisible by the specified length, the last part of the line might create a line shorter than the specified length.

The file is going to be saved with the **Do not write Newline delimiters** option enabled whether the original file contained Newline delimiters or not.

3. Click the **Open** button to open the file on the host. Qedit will use your existing connection if you have one. Otherwise, it will establish a new connection.

Opening Files with Binary Data

Qedit can edit files that contain text information. It can also edit a binary file, although Qedit will likely corrupt these files so badly that you might not be able to touch them with the application that created them.

That is the case for documents created by word processing applications such as Microsoft Word. Qedit can open these files and display their contents as best it can. The document window, however, will probably contain all kinds of garbage characters that will not make any sense. ***Do not save the file, even if you have not changed anything.***

Server Warnings

The server might detect inconsistencies when opening host files and send warning messages to the client. The file is going to be open but might be different from what is the host. The user should then check the file and fix the problems before accessing them again.

Lines Exceeded Maximum Length

Some files must have specific attributes based on the language they correspond to. One of these attributes is the **Record Length**. For example, COBOL source files are supposed to have fixed-length records of 74 bytes (Cobol) or 80 bytes (Cobx). If one or more lines exceed that size, Qedit displays the following warning:

```
Some lines exceeded the Language maximum line length. These lines have
been folded. See line 15 (Server warning 1)
```

The server folds the overflowing lines. This means the server puts the characters exceeding the **Record Length** onto a separate line. Effectively, this means the file appears to have more lines than the original. The warning message shows the number of first folded line (15 in the example above). The user should examine that line and determine if the file can be fixed and saved without losing or damaging the original file.

Server Ran Out of Line Numbers

The server requires line numbers to be able to edit host files. If the file does not already contain numbers, the server assigns them as it reads the file. It determines as precisely as possible the range and increment to use. For extremely large files, it is possible for the server to run out of line numbers.

Other files such as COBOL source files might already contain line numbers. In this case, the server uses these numbers. The server makes sure that the line numbers are valid (all numeric) and in ascending sequence throughout. If it encounters an invalid

line number, the server tries to recover the error and starts assigning line numbers. This could also cause the line number to exceed the absolute maximum.

In these situations, Qedit displays the following warning:

The server ran out of line numbers while texting the file. The file may be too large or contain invalid line numbers. See line 032980. (Server warning 2).

When the server runs out of line numbers, it simply stops reading the file. This means the file as seen in Qedit for Windows does not have all the original lines. The user should **close the file immediately without saving it**. The file should be examined closely to determine the cause.

The possibly invalid line number is displayed as part of the warning message after "See line". In order to find the corresponding line, you should close the file without saving and open it with the **Force Unnumbered** option enabled.

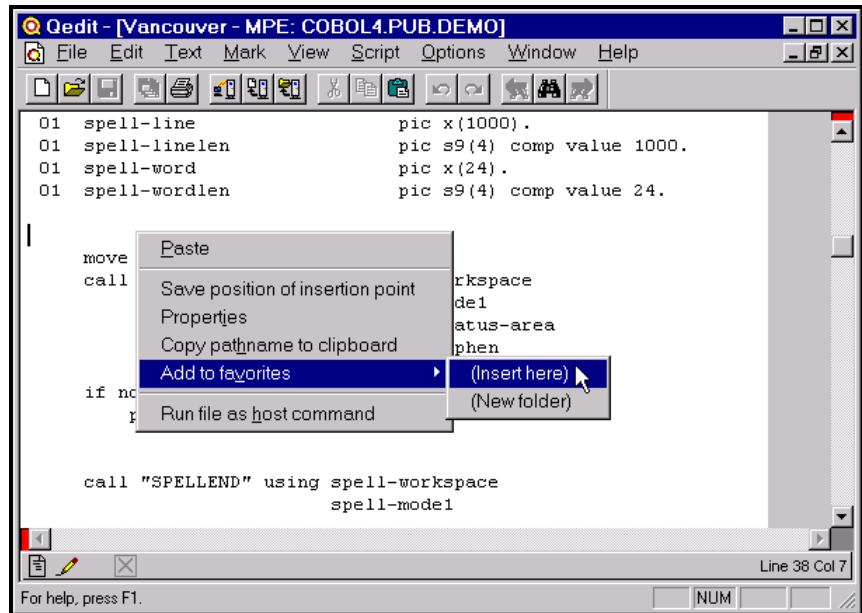
Maintaining and Using a List of Favorites

The **Favorites** command on the **File** menu allows you create and maintain a list of files that you access most often. You can add filenames to the list. You can create groups to assemble files related to a project for example. Of course, you can remove them from the list. A file can only appear once in the list even if you use folders.

Adding to the Favorites

To add a file to the list of favorites:

1. Open the file
2. Click the right mouse button to display the shortcut menu
3. Click on **(Insert here)** to add the file at this level



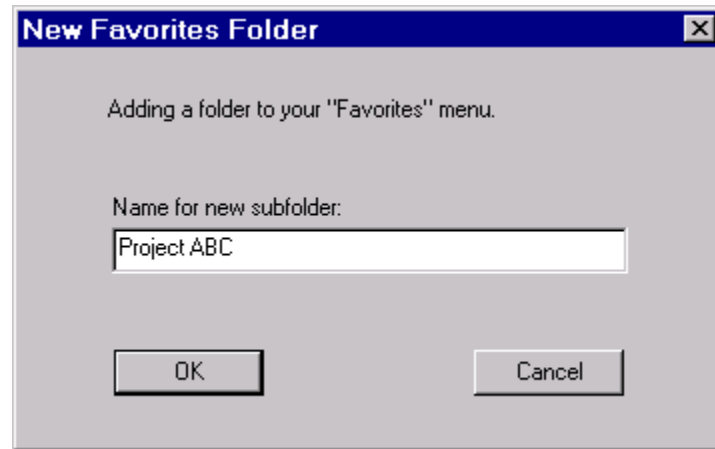
Adding a file to the top level Favorites

Adding to a Favorites Folder

You can organize the Favorites list into groups or folders. A folder could represent files needed for a larger project. You can also have multiple levels of folders. In other words, a folder can contain other folders.

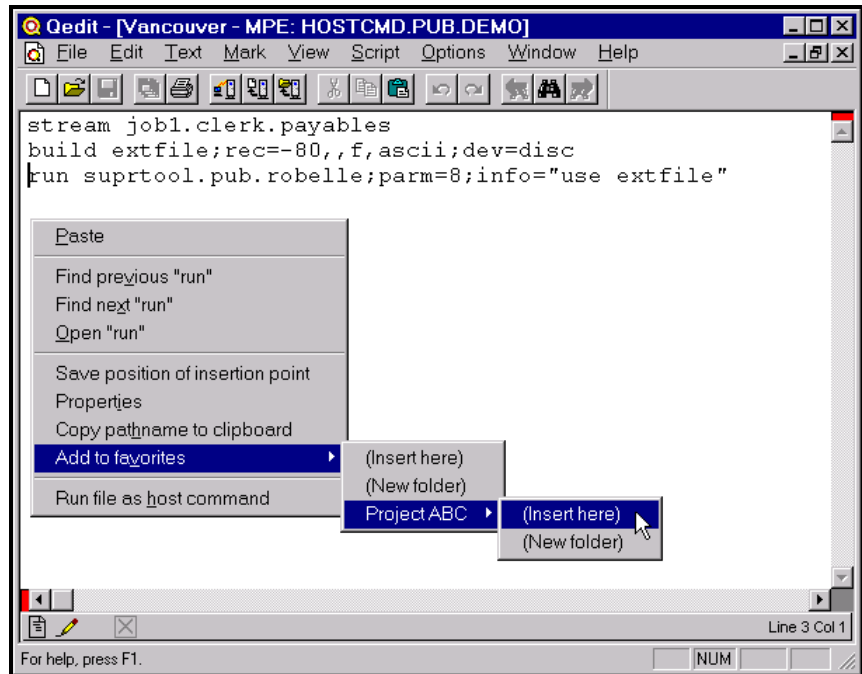
To create a new folder and add a file to it:

1. Open the file
2. Click the right mouse button to display the shortcut menu
3. Click on **(New folder)**. This should bring the **New Favorites Folder** dialog box.
4. Enter the name of the folder. The name can contain any character except a colon ":". It can be as long as needed.
5. Click **OK**. The folder is created and the filename is added to it.



Creating a new Favorites folder and adding a file to it

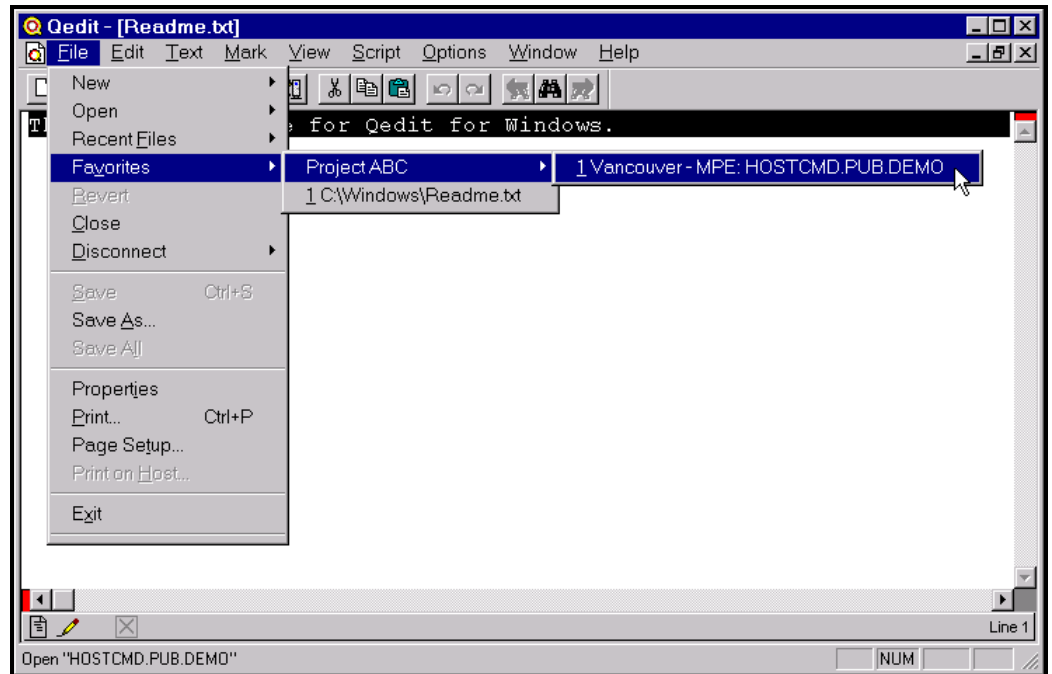
If the folder already exists, it's going to appear on the shortcut menu. To add a file to a folder, select the corresponding (Insert here) command.



Add a file to an existing Favorites folder

Opening a File from the Favorites

The filename now appears on the **Favorites** list. To open it, select the **Favorites** command on the **File** menu, find the file in the list and click on it. Because filenames can be very long, the name may be shortened in the list. Qedit replaces some characters in the middle with three dots. However, when you put the mouse pointer on a name, Qedit displays the fully-qualified name on the status bar.



Opening a file from the Favorites

There's no limit to the number of files that can be added to the Favorites list and each folder. However, Qedit only displays a limited number of files. This limit can be changed in the **General tab of the Preferences dialog**. You can have between 5 and 50 files. The default limit is 20.

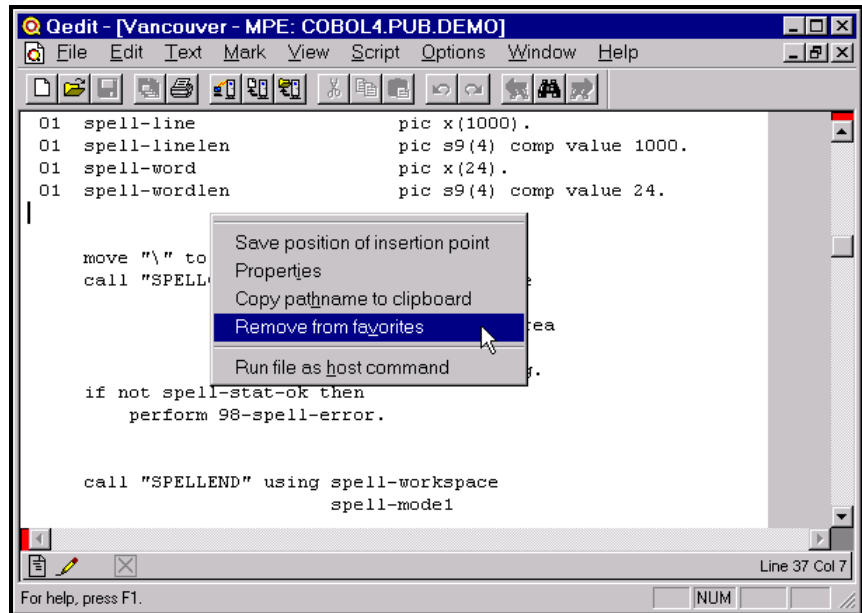
When the filelists are displayed, a shortcut character is assigned to the first 35 files. This means you can select a file by combining the **ALT** key with corresponding character as in **ALT+3** or **ALT+B**. The first are numbered from 1 to 9. The next 26 are labeled from A to Z. Files 36 through 50 do not have a shortcut character.

Removing from the Favorites

To remove a file from the favorites list:

1. Open the file
2. Click on the right mouse button to display the shortcut menu
3. Select **Remove from favorites**

You don't have to find it in the list. Since it can appear only once, Qedit is able to identify it unambiguously and remove it.



Removing a file from the Favorites

Viewing the List of Open Files

Shortcut: ALT+W

If you have opened more than one file, you can view your list of open files by selecting the **Window** menu. The opened files and newly created files will be listed at the bottom of the menu list, and they will be numbered 1, 2, 3 etc. You can go to any of the files by typing its number or by selecting the file from the menu list.

Customizing Opening a File

Setting Default Options for New Files

If you want to set the default options for a new file,

1. Select the **Preferences** command from the **Options** menu.
2. When the **Preferences** dialog box appears, go to the appropriate tab for the platform where the file resides. The platforms currently supported are UNIX, MPE and local files.
3. When you see the correct tab, select the options you want to use as defaults.
4. If desired, you can set the defaults for the other platforms at this point also.
5. Click **OK** to save the options as the new defaults.

Default options are used for new files, files that are not in the document database i.e. have never been opened in Qedit or files in the document database but for which you have not specified custom settings.

Changes made in the **Preferences** dialog box are not applied to currently opened documents. Most of these options are described in detail in "Defining File Options" on page 63. To use a default color scheme, select the scheme name from the **Default**

colors list box. If you want to define a new color scheme or change an existing one, read the "Setting the Default Color Scheme" on page 121.

Loading Local Files into Memory

This option is only available in version 4.7.04 and later.

The **Load local files smaller than ... K into memory** option lets you control whether Qedit will load local files into memory. When this option is disabled (default), Qedit brings only the necessary parts of the file into memory. As a result, you can edit local files that are much larger than your PC's memory

If this option is disabled, Qedit keeps the original file open during the editing process and prevents other people from modifying the file at the same time. When Qedit saves your changes, it merges the modified lines into the original file.

Reading the entire local file into memory, however, has other advantages. Editing is very fast because the text you work on is always present in memory. By releasing the file as soon as Qedit has finished loading it, the file becomes available to other people almost immediately and stays available until Qedit reopens it to save changes.

To specify which local files get loaded into memory,

1. Select **Preferences** from the **Options** menu.
2. When the Qedit **Preferences** dialog box appears, select **Load local files smaller than ... K into memory** and specify a threshold file size. If a file is smaller than or equal to the specified value, Qedit will load the whole file into memory. If the file is larger, Qedit will load only the necessary parts of the file, which is how Qedit works when this option is disabled.

The minimum value (default) is 32K. The maximum value is 4,096K. For files below the configured threshold, Qedit allocates a memory block based on their size. If you open several large local files at the same time, you can quickly consume available memory. For this reason, we recommend that you use a conservative value for a threshold.

3. Click **OK** to save the option settings and close the dialog box.

If Qedit has difficulty loading a file because you don't have enough memory, the result of an open operation is unpredictable.

This option also affects save operations. For more details, see "Saving & Closing Files" on page 157.

Typing, Selecting & Editing Text

Overview

Typing and editing features are the foundation of Qedit for Windows. These features are supplemented with various ways to select text and to handle nonprinting, control and international characters. Qedit gives you the ability to cut, copy and paste within a file or between files. And if you decide not to keep your changes, you always have the option to undo or redo your work.

Determining Your Position in a File

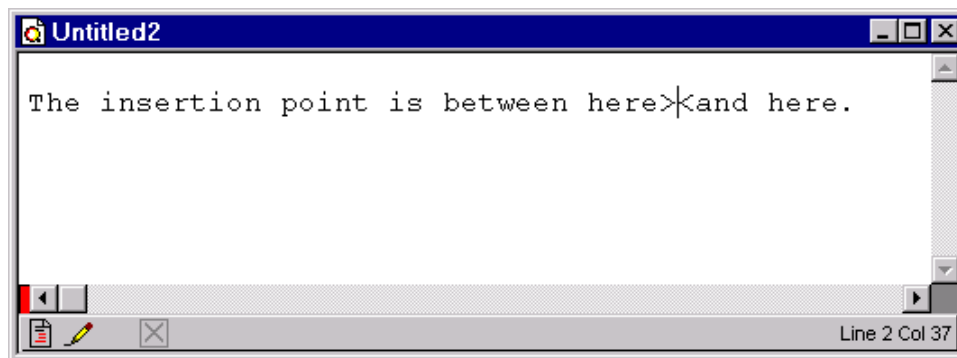
For typing and editing, you need to know your exact position within a document. The status bar indicates the current position of your cursor on the page. See “Document Status Bar” on page 33 for more details.

Cursor Position and Appearance

The current location of the cursor is called the insertion point. Qedit for Windows uses a blinking cursor to show the current insertion point. The cursor moves around when you press a cursor movement key or click the mouse button.

If a block of text *has not* been selected, the cursor position appears on the document status bar as a relative record and column coordinate. If a block of text has been selected, the coordinates of the cursor appear as the starting line and column, and the ending line and column.

The vertical bar displayed between the greater-than and less-than characters is not part of the text. It is the insertion point indicator.



Blinking vertical bar indicates the insertion point

Controlling the Insertion Point Movements

In some applications, the SCROLL LOCK key is used to control the behavior of the insertion point as you move around in a document. Even though Qedit displays the current status in the toggle key box of the status bar, Qedit does not work any differently whether the key is enabled or disabled.

If a block of text has not been selected, you can change the behavior of the insertion point by changing the status of the **Move caret when navigating** option on the **General** page of the **Preferences** dialog. When you start Qedit for Windows, this option is enabled. As you move through a file, the insertion point stays in the same position relative to the screen, even though its actual line number changes.

If you disable **Move caret when navigating** when text is not selected, the cursor stays in the same line and column position, even if you go to another part of the file. As a result, the insertion point can scroll off the screen if you move too far away from it. To quickly return to the insertion point, just press a cursor movement key. For more information, see “Using Cursor Movement Keys” on page 137.

If you select a block of text, the behavior of the **Move caret when navigating** option does not change. Enabled or disabled, the insertion point stays in the same line and column position, and can scroll off the screen if you move too far away from it.

Insert Mode vs. Overwrite Mode

Qedit has two modes of behavior when you add text: insert mode and overwrite mode. The active mode is shown on the status bar of the main window. You can change the mode by pressing the INSERT key.

When you start Qedit for Windows, it is in insert mode. When you add text at the insertion point, which appears as a blinking vertical bar, Qedit moves existing text to the right. For insert mode, there is no indicator in the status bar of the application window.



Toggle keys status while in insert mode

When you change to overwrite mode, the cursor becomes a blinking rectangle or block. In this mode, the text you type overwrites the existing text. This is equivalent to deleting the existing text and replacing it with new text. In overwrite mode, the BACKSPACE key does not remove characters as it moves to the left; it becomes equivalent to using the LEFT ARROW key.

For overwrite mode, **OVR** appears in the status bar of the application window.



Toggle keys status while in overwrite mode

Typing Text and Inserting Special Characters

Qedit for Window's flexibility with typing text and inserting special characters gives you the ability to address an international audience, insert control characters and add regular text, all in the same file.

Typing Text

If you open a new document, you can start typing at the top of the file.

If you want to insert text into an existing document, you can select the insertion point for the new text. To type text into an existing document,

1. If you do not want to overwrite the existing text, make sure you are in insert mode. See "Insert Mode vs. Overwrite Mode" on page 88.
2. Position the insertion point where you want to begin typing.
3. Enter the text you want to add.

In a new document, use **Save as** to assign it a file name. Otherwise, use the **Save** command to save your changes.

Inserting Control Characters

The character sets for Windows, MPE and UNIX include the 32 characters in the ASCII character set that are control functions. For example, character value 7 in the ASCII character set means ring the bell.

The host-based method of using CTRL+G does not insert control characters into a file because this shortcut combination is used for Qedit's **Find again** function. As a result, you need another way to insert a control character. Windows does not have a standard way to handle control characters, and Qedit for Windows cannot do this either. There is no keyboard shortcut for inserting special characters.

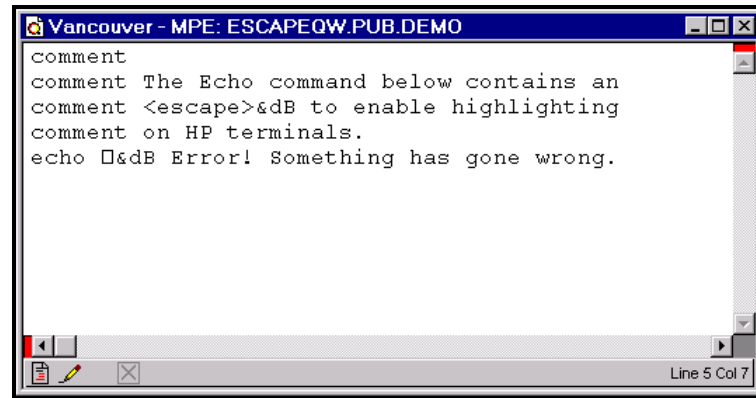
Check Set Decimal On, Change, or QZMODIFY commands in host-based Qedit.

Host-based Qedit, however, can handle these characters. The workaround is to save the file on your host, edit it with host-based Qedit and use one of the methods available on the host.

Inserting Nonprinting Characters

Nonprinting characters are special characters whose ASCII values (e.g., 0 to 31, 127, etc.) cannot be displayed in a recognizable form. Within most applications these characters have very specific uses, including carriage return, line feed, etc.

In Qedit, nonprinting characters appear as small square boxes within the text.

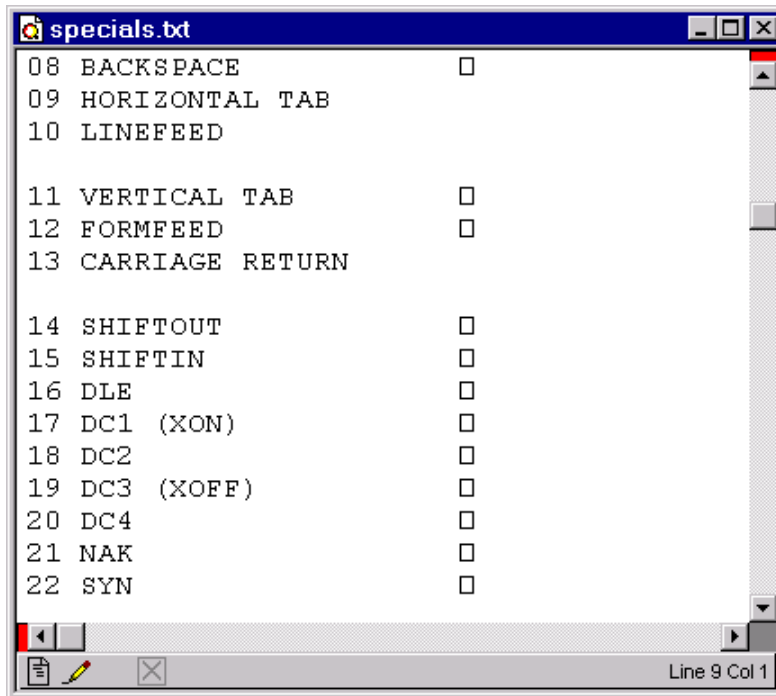


Displaying nonprinting characters

You cannot insert or modify nonprinting characters directly with Qedit for Windows. If a file contains these characters, Qedit does not change them. You can work with these characters, however, if you use another program (host-based Qedit for example) to create a character library file, so to speak.

This workaround does not work with ASCII 9, the TAB character, because Qedit uses tabs for formatting purposes.

For example, you can call this library file `Specials` and keep it somewhere on your PC or host. Each line contains a number that represents the ASCII value followed by a space and the corresponding nonprinting character. If you need to enter a special character into a file, such as the escape character (ASCII 27), just open `Specials`. Then highlight the square box next to the ASCII value that you need and copy it to the Clipboard. When you return to the file you are currently editing, paste the character at the desired location.



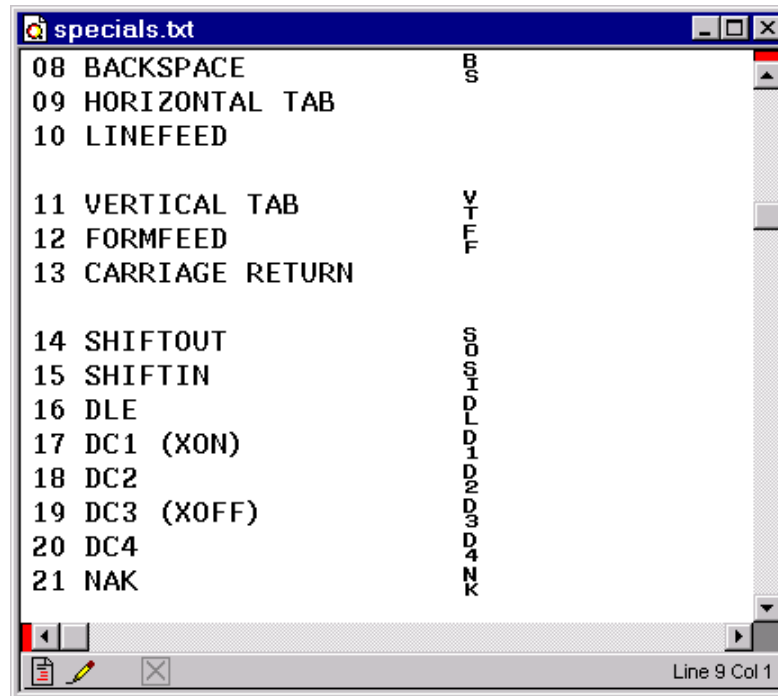
File with nonprinting characters ready for copy

If you need to work with terminal escape sequences, you can have separate lines just for them. For example, you can define a few lines for inverse video, blinking characters, inverse video off, etc.

Nonprinting Characters for Reflection Users

If you have the Reflection terminal emulator from WRQ, Inc., you can use one of the fonts provided with the software to display nonprinting characters. You can create the Specials library file as described in "Inserting Nonprinting Characters" on page 90. With the library file open, change the display font to `r_ansi`. This displays most nonprinting characters as two tiny characters on top of each other. For example, a bell character appears as a small *b* over a small *l*, an escape character as a small *e* over a small *c* and so on. You still cannot enter these characters, but, at least, you can see what they are.

Qedit does not display line formatting characters such as carriage return, line feed and horizontal tabs; it interprets them instead.



Nonprinting characters with the r_ansi Reflection font

Inserting Special and International Characters

If the keyboard for your PC is not a standard English keyboard, you just press the special keys on your keyboard to insert the characters.

With Windows 95, you can still insert special characters into your document with a keyboard for another language. See "Special and International Characters" on page 49.

Selecting Text

In the context of Qedit, the term “selecting” is synonymous with “highlighting.” If text is highlighted, it appears on your screen in inverse video.

If you don’t highlight a string of text in your file, the only edit operations that are available to you are to type, backspace or delete a single character. The key to more powerful functions is text selection.

Here is a partial list of what you can do with the selected text:

- delete it by pressing the DELETE or BACKSPACE (in insert mode) key
- replace it by typing some new characters
- copy it to the Clipboard (Toolbar: **Copy** button)
- cut it from the document and write it to the Clipboard (Toolbar: **Cut** button)
- if the selected text is an embedded file name, open the file

*Shortcut: CTRL+C or
ALT+E,C*

*Shortcut: CTRL+X or
ALT+E,T*

*Shortcut: CTRL+D or
ALT+F,O*

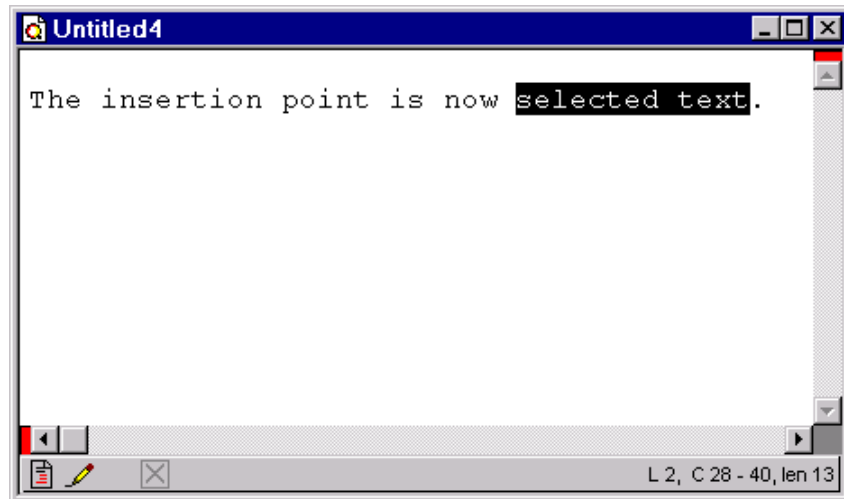
*Shortcut: CTRL+H or
ALT+E,F*

- search for the selected text (Toolbar: **Find** button)

Cursor Position and Appearance During Selection

If you select some text, the selection replaces the cursor and becomes the current insertion point. If you start typing, the selected text is replaced with the first character you type. If you continue typing, the new text either pushes the existing text to the right or overwrites it, depending on whether you are in insert or overwrite mode.

Any selected text automatically becomes the insertion point. The cursor is not displayed during selection.



Selected text is the insertion point

If you use the scroll bar to move to another part of the document, the selected text continues to stay selected, even though it might not be visible on your screen. The selection is lost only when you move the cursor to a new location. If you forget that you selected text and start typing in another location, you will return to the selected text because that is the current position of the cursor.

Multiple Ways to Select Text

Qedit for Windows gives you numerous ways to select text. You can use your mouse to select characters, words, paragraphs or complete sections. You can also use the **Find** operation or the SHIFT key in combination with cursor movement keys to make your selection. Qedit's flexibility lets you choose the method that serves you best.

Selecting with the Mouse

A common method to highlight text is to use the mouse. As in many other applications, this method is a standard feature of Windows. See "Mouse Operations" on page 38.

Selecting with the Shift Key

Many times you are busy editing text and realize you want something highlighted. If you move your hand from the keyboard to use the mouse, you slow down your thinking and editing process (at least if you are a touch typist). What can you do?

There is a way to select text without moving your hands from the keyboard: the SHIFT key in combination with cursor movement keys. When you use the SHIFT key, the text is selected as the cursor moves.

Some of the most useful combinations are SHIFT+HOME to highlight to the start of the current line, SHIFT+END to highlight to the end of the current line, and HOME, SHIFT+END to highlight the current line.

If you want to select all the lines from the cursor to the beginning of the file, use CTRL+SHIFT+HOME. If you want to do the opposite (i.e., select all the lines from the current location to the end of the file), use CTRL+SHIFT+END. If the current selection is rectangular (CTRL+LEFT BUTTON) and you use the key combinations above, the selection is extended accordingly and the selection remains rectangular. In other words, this is an easy to select columns throughout the file. For more information, see “Using Cursor Movement Keys” on page 137.

If you use the cursor movement keys to move away from the selected text, the selection is canceled. You can also use the ESCAPE key to cancel the current selection. Using these keys is good practice because it prevents you from accidentally deleting the selection, especially if you are not going to use a selected string after a **Find** operation.

Selecting with a Find Operation

Just as in other Windows editors, Qedit also highlights text when it performs a successful **Find** operation. This not only makes it easy to see where the text was found, but it makes it easy to change or delete the selected text. Perhaps too easy! If you type a character, or press the DELETE or BACKSPACE key after a successful **Find** operation, you can accidentally delete or modify the selected text.

Copying, Moving and Pasting Text

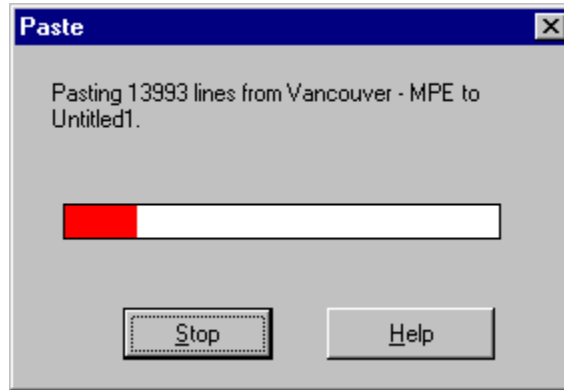
The Clipboard

The Clipboard is a conceptual entity that can be used for temporary storage—a place where you can save lines of text, a picture or some sound, then use it again later. Just as in other Windows program, you cannot append to Qedit’s Clipboard. That means the contents of the Clipboard are erased as soon you copy something new to it. You can use the Clipboard to copy text to a Qedit document or to another application. The Clipboard is closed when you shut down your system.

You can cut or copy to, and paste from the Clipboard. If you have already used **Cut**, **Copy** or **Paste** in Notepad or Microsoft Word, you will not have any trouble with the Qedit Clipboard. It works the same way, except for only one difference: the Clipboard might not actually reside on your PC when you edit a file on a host system.

When you cut or copy from a host file to the Clipboard, Qedit does not transfer this data to the PC, unless you paste it into a file on another host or into a local file. There are two reasons for this: (1) it could take a long time if the file is very large and (2) you are quite likely to paste it back into another file on the same host system! Instead, Qedit copies the selected text to the internal Hold file on the host server. Unique to each logon, this Hold file functions much like the Clipboard.

Qedit copies the data from the host to the PC only when you paste to a local PC file or to a file on another host. For example, when you cut or copy on one connection and paste on another one, Qedit must physically transfer the Clipboard from the first host to the PC, then to the second host. This can take a while, so a small progress indicator window is displayed during this process. The same process occurs if you cut or copy from a host file and then paste into a local Qedit file.



Paste progress indicator dialog box

Workarounds to Clipboard Cache Problem

If you cut or copy a host file, then paste it into a PC application, such as Notepad or Microsoft Word, you could run into one problem.

If the Clipboard lines fit into the Qedit cache, the paste request will be satisfied. If the Clipboard lines do not fit into the cache, Qedit cannot get the missing lines by issuing requests to the host, which makes the Clipboard appear empty to another PC application. If you run into this problem, you have two options:

1. You can copy the text in chunks. For each chunk, copy it to the Clipboard so that it is effectively brought into cache. Then you can paste it into the target application.
2. You can also force Qedit to put all the lines that you select into the document cache. First choose **Copy Special** from the **Edit** menu. This brings the following dialog box:



Copy Special dialog box with Copy to local clipboard option

Enable **Copy to local Clipboard** and select **Copy**. Once the copy operation is completed, the selected lines are available on the Clipboard.

There is no preset limit to the size of the local Clipboard. The amount of text that can be copied to the Clipboard can be different for each user because it is determined by the PC configuration.

If you have to deal with tab characters in files on different platforms, you can use the **Expand tabs while copying option** which is explained in "Expanding Tabs in Clipboard" on page 102.

Copying Text

If you want to copy some text, you can easily store it on the Clipboard. To copy text to the Clipboard,

Shortcut: CTRL+A

Shortcut: CTRL+C

Toolbar: 

1. Select a contiguous block of text by highlighting it. This could be as small as one character or as large as the entire file. For example, use the **Select All** command of the **Edit** menu to select the entire file.
2. Select the **Edit** menu and click **Copy** to copy the selected text to the Clipboard. The previous contents of the Clipboard are replaced with your new selection. The text in your document is still selected.

Pasting Text

If you cut/copy and paste in the same host document or another host document open on the same connection, Qedit issues a special command to the server to have the **Paste** operation done on the host. The only lines transferred to Qedit on the PC are the updated lines visible on the screen.

If you are working with lines of the same length, there is no problem. If you are copying lines that might exceed the record length of the target file, Qedit then has to make some decisions. Because local files, MPE host files and UNIX host files work differently, the result of a paste might be different.

Source and target files on the same connection.

MPE host files and, to some degree, UNIX files typically have a maximum record length as defined in the **File options** dialog box. If you are copying lines between two (2) files on the same connection, the operation is done by the server. Because the server knows the characteristics of the source and target files, it can fold lines before inserting them in the target file. Folding means Qedit puts as many characters as it can on a line and moves the rest to a separate line.

Source and target files on different systems.

In all other cases, Qedit truncates lines that are too long. If this happens, Qedit displays a warning message.

To paste text in a file,

Shortcut: CTRL+V


Toolbar: 

1. Position the insertion point where you want to paste.
2. Select **Paste** from the **Edit** menu to insert the contents of the Clipboard to the file.

Moving Text

The process of moving text consists of two parts. First you must remove the text from its original position, then you have to paste it to a new location. To move text,

Shortcut: CTRL+X

Toolbar: 

1. Select the block of text you want to move by highlighting it.
2. Select the **Edit** menu and click **Cut** to put the selected text on the Clipboard. The previous contents of the Clipboard are replaced with your new selection. The text in your document is still selected.

Shortcut: CTRL+V

Toolbar: 

3. Position the insertion point where you want to insert the text. This could be in the same file, another file in Qedit or even another application. Select the **Edit** menu and click **Paste**.

You can also perform operations to and from the Clipboard using shortcut menus. See “Shortcut Menu on Document Workspace” on page 45.

Deleting Text

There are several ways to delete text in Qedit for Windows. If you select text before deleting it, see “Selecting Text” on page 92 for more information.

1. The first and most obvious way to delete text is by pressing the DELETE key. This deletes the single character that is immediately to right of the cursor, which might be a space character or a TAB character. An interesting case is when you go to the end of a line and press DELETE. You remove the NEWLINE character between this line and the next, effectively appending the next line to the current line.
2. If you are in insert mode, you can press the BACKSPACE key. This key deletes the character before the cursor position, effectively moving you back one character and removing it at the same time. If you use the HOME and BACKSPACE keys, you remove the NEWLINE character and append the current line to the previous line.
3. You can highlight some text (any sequence of contiguous characters including NEWLINE characters) and then press the DELETE or BACKSPACE key. The highlighted characters are deleted and the remaining line fragments to the left and right are combined.
4. You can also highlight some text and then press the **Cut** command of the **Edit** menu. This deletes the highlighted text and writes it to the Clipboard. You can get the deleted text back with the **Paste** command in the **Edit** menu.

The BACKSPACE key does not have any effect in overwrite mode. The cursor just moves to the left.

Merging Two Lines

Remember that if you delete the end of a line and remove the NEWLINE character, you will merge two lines into one. If the new line exceeds the file’s maximum line length, which is determined by the **Language** option, you will get an error message that says, “The resulting line would be too long.” If you need to remove the NEWLINE character, you will have to break one or both lines into shorter lines before you will be allowed to merge the pieces together.

Working with Columns and Tabs

Server version 4.8 or higher is required.

Qedit provides a number of commands to easily deal with text in specific columns. Some of the commands are in the **Column** submenu of the **Edit** menu and others are in the **Text** menu. The **Find** and **Replace** dialog boxes also have features that help you deal with columnar data.

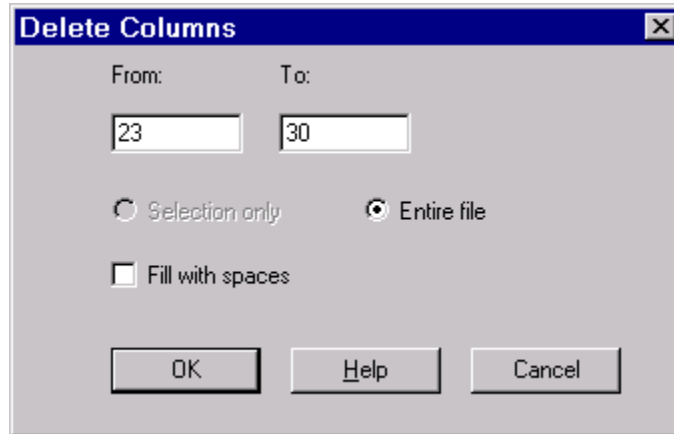
Throughout these menus, column numbers are relative to the **Language** of the file. For most files, the first column is one. For some languages, such as COBOL, the

first column is seven. Values in text boxes must be entered accordingly. For example, if you want to delete the first two columns of a COBX file, you have to enter seven and eight as column values.

Deleting Text in Columns

Shortcut: ALT+E, O, D

If you want to remove or replace text between two columns, use the **Delete** command of the **Edit** menu.



Delete Columns dialog box

Enter the starting column number in the **From** text box. If you do not specify a column number in the **To** text box, Qedit assumes you only want to work on one column. If you want to remove more than one column, simply enter the column number of the last column in the **To** text box. If you put the cursor in the **To** text box, Qedit automatically fills it with the file's **Record Length**. This is useful if you want to clear all the lines from one point to the end of the line.

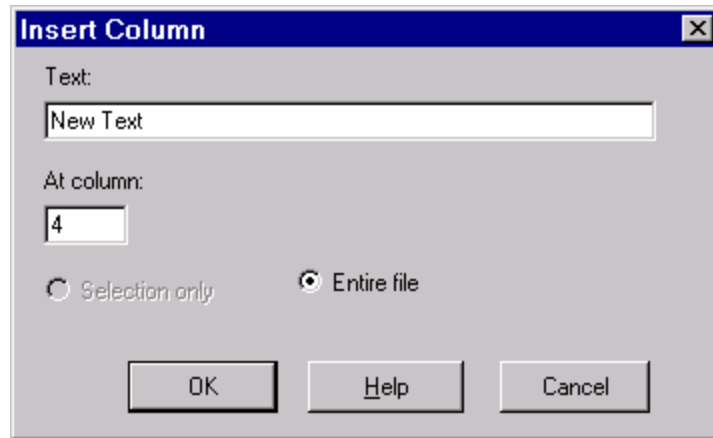
If you do not have an active selection, the modification is applied to the whole file as indicated by the **Entire file** option. If you have selected some text, the **Selection only** option is enabled and the operation is applied only to the selected text.

If the **Fill with spaces** option is disabled, Qedit removes the text in the specified columns. Text to the right of the **To** column is then shifted left to fill the vacant columns. If **Fill with spaces** is enabled, Qedit removes the text in these columns and replaces it with spaces. This preserves the text alignment to the right of the columns.

Inserting Text in Columns

Shortcut: ALT+E, O, I

If you want to insert text at a certain column position, use the **Insert** command of the **Column** submenu.



Insert Column dialog box

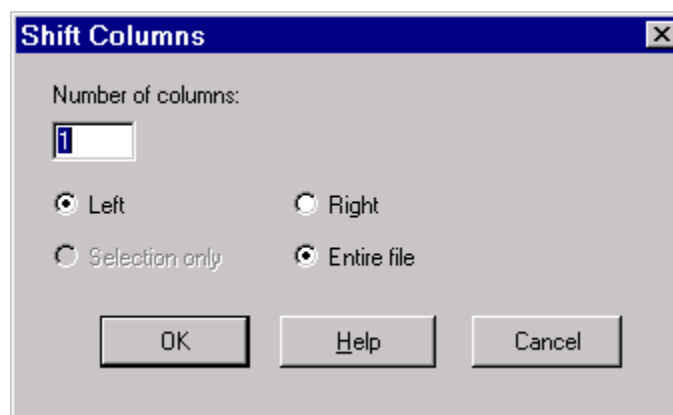
Type the text you want to insert in the **Text** text box. In the **At column** text box, enter the column at which you want the text to appear. The specified string is inserted and text to the right of the column is shifted. If there is not enough room on the line to accommodate the new string, an error is reported and the operation is stopped.

If you do not have an active selection, the modification is applied to the whole file as indicated by the **Entire file** option. If you have selected some text, the **Selection only** option is enabled and the operation is applied only to the selected text..

Shifting Text in Columns

Shortcut: ALT+E, O, S

If you want to shift text left or right, use the **Shift** command of the **Column** submenu. When shifting lines to the right, assuming the **Use spaces for tabs** option is disabled, a tab character is inserted at the tab stop to the left of the first nonblank character on the line. This is to preserve the tab stop relative alignment of lines that do not start on a tab boundary. If **Use spaces for tabs** is enabled, spaces, instead of a tab character, are inserted at this location.



Shift Columns dialog box

In the **Number of columns** text box, specify the number of columns by which you want to shift the text. Select **Left** if you want to shift the text in that direction, which is similar to deleting a number of spaces at the beginning of a line. Select **Right** to

move the text in that direction, which is similar to inserting a number of spaces at the beginning of a line.

If you do not have an active selection, the modification is going to be applied to the whole file as indicated by the **Entire file** option. If you have selected some text, the **Selection only** option is enabled.

To shift entire lines left or right, see "Shifting to the Next Tab Stop" on page 101.

Shifting to the Next Tab Stop

If you want to quickly align text to the next tab stop, you can use the **Shift left** or **Shift right** commands on the **Text** menu. You have to set some tab stops in the file to use these commands. The shifting is done at the beginning of the line. If the **Use spaces for tabs** file property is enabled, Qedit inserts space characters in column one. If **Use spaces for tabs** is disabled, Qedit inserts a tab character in column one.

Shortcut: ALT+T, L

Shortcut: ALT+T, R

This is not allowed if there is an active rectangular selection.

If you use **Shift left**, the text moves to the previous tab stop on its respective line.

If you use **Shift right**, the text is moved to the next tab stop on its respective line.

If the text cannot be shifted further, Qedit simply stops shifting. For example, if you keep using **Shift left**, all the lines would eventually be left-justified in column one.

For a faster way to perform the same operation, select the lines (completely or partially) you want to shift and

- Use the TAB key to shift to the right
- Use SHIFT+TAB to shift to the left

Warning: this behavior is different than in previous versions.

This feature introduces an incompatibility with previous versions. If you intended to replace the selection with a tab character, pressing the TAB key now shifts the lines to the right. The text is not replaced. If you want to replace the selection, you have to use the DELETE key (or any other key except TAB) to get rid of the selection, then enter the tab character.

Removing or Inserting Tab Characters

You can visually align text in files by using spaces or tab characters. Spaces viewed on the screen are stored in as many characters in the file. If you type five spaces, the file stores five bytes. Tab characters however, take up only one byte in the file no matter how it appears on the screen. Using a mix of spaces and tab characters can be deceiving. The size of a file changes depending on which one you use. Since these are different characters, programs that read the file must be able to handle both.

The Detab and Entab commands work only on complete lines.

If you need to use either formats, you might have to replace one with the other. This can be very tedious. Qedit has the ability to do the replacement with a single command from the **Text** menu.

If you have a file containing tab characters and you want to use spaces instead, use the **Detab** command. Qedit preserves the visual alignment, but it physically removes all the tab characters and replaces them with the appropriate number of spaces.

The opposite is also true. You can use the **Entab** command to convert spaces into tab characters. Qedit looks at the text and the current tab stops. When it detects the possibility of replacing one or more spaces by a single tab, it does so. In addition,

Qedit tries to preserve the visual alignment. If need be, it will combine tab characters and spaces.

Expanding Tabs in Clipboard

When working with files on different platforms, it is sometimes annoying to deal with tab characters. If you are working with a local file and you want to make sure the text sent to the Clipboard does not contain tab characters, simply enable the **Expand tabs while copying** option in the **Copy Special** dialog box.



Copy Special dialog box with Expand tabs while copying option

This dialog box can also be useful when copying large amount of data from a host file to a local file. See "Workarounds to Clipboard Cache Problem" on page 95.

Working with Rectangular Selections

Server version 4.8 or higher is required.

You can select text in specific columns instead of complete lines or a selection spanning multiple lines with different start and end columns. All these selections basically work the same. However, there are differences specific to rectangular selections.

Copying and Cutting a Rectangular Selection

To copy or cut text from certain columns, all you have to do is select the area. You can create rectangular selections only with the mouse. There is currently no equivalent keyboard function. To create a rectangular selection, simply hold the CTRL key and click the mouse button as you would for a regular selection. Once the text is selected, use the **Copy** or **Cut** command on the **Edit** menu to send the selection to the Clipboard. You can also use the CTRL-C and CTRL-X shortcut keys.

Pasting a Rectangular Selection in Replacement

To paste the contents of a rectangular selection, you have to select the area into which you want to paste. This area does not have to be exactly the same size as the area copied to the Clipboard. Qedit does its best to fit it in the destination.

If the destination area is narrower than the source selection, wider lines from the Clipboard are truncated.

If the destination area is wider than the source selection, text from each source line is pasted on a separate line and Qedit pads the rest of the area with spaces.

If the destination area is shorter (i.e., has fewer lines than the source selection), the extra lines in the Clipboard are dropped. For example, if the source selection had 10 lines and the destination has only six, the last four lines in the Clipboard are not pasted.

If the destination area is longer (i.e., has more lines than the source selection), text from the Clipboard is repeated. For example, if the source selection had five lines and the destination area has 10 lines, the source data appears twice. If you want to get rid of the redundant data, you can select it and use the **Clear** command on the **Edit** menu. This replaces the contents of the selection with spaces. If you want to remove the text and shift the remaining text, use the DELETE key.

Inserting a Rectangular Selection

There is currently no direct way of inserting a rectangular selection. If you want to do this,

1. Select the lines on which you want to insert the Clipboard.
2. Use the **Insert** command of the **Column** submenu on the **Edit** menu to create an empty area. Make sure you make it wide enough to accommodate the text you copied to the Clipboard.
3. Select the newly created area, if it's not already selected.
4. Paste the Clipboard.

Deleting or Clearing a Rectangular Selection

To remove the contents of a rectangular selection, use the DELETE key. This removes the selection and shifts the rest the lines to the left. To clear the contents of the selection (i.e., replace it with spaces), use the **Clear** command on the **Edit** menu.

Extending a Rectangular Selection

To select columns from the current location to the beginning or end of the file, use CTRL+SHIFT along with HOME or END respectively. For these combinations to work, the current selection must be rectangular already i.e. selected with CTRL+LEFT BUTTON. The CTRL+SHIFT combination actually extends the selection in the requested direction.

Undoing and Redoing

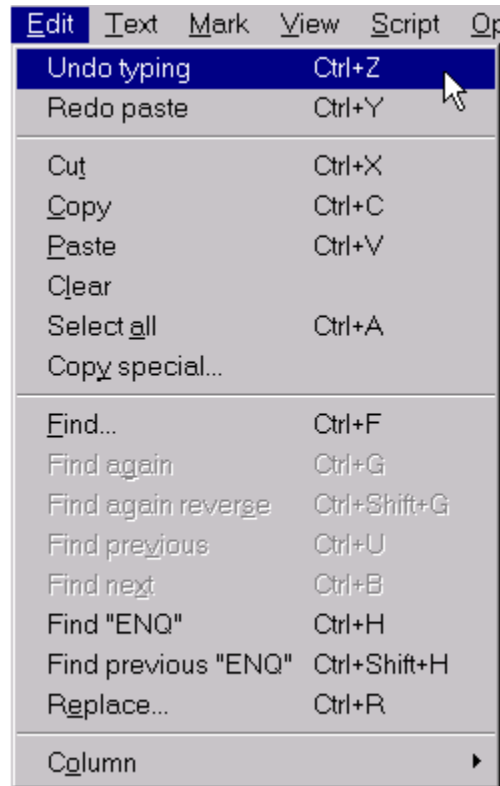


Undo and Redo tools

The **Redo** and **Undo** commands, which are available on the **Edit** menu, repeat or cancel the last modification. You cannot undo or redo changes out of order. If you have performed a number of operations, the **Redo** and **Undo** commands apply only to the last one.

When you have multiple files open, Qedit maintains a separate undo transaction log for each file. If you cut and paste from one edit window to another, these are treated as *independent* transactions. That is, canceling the paste command into window A **does not ensure** that the corresponding cut command from window B is undone. As a result, the cut text could be lost.

The descriptions in the **Undo** and **Redo** commands on the **Edit** menu change according to the last transaction. At the start of a session, the **Undo** and **Redo** commands are not available (menu shows **Can't undo** and **Can't redo**) because there have been no previous transactions. As you change the file, the description of the last operation is posted to the menu. For example, if you enter text in a couple of different places, then undo the last operation, the **Edit** menu will look like this:



Edit menu after a Redo command

The description shows “typing” as the last edit operation processed by both **Undo** and **Redo**. If the last edit operation had been **Paste**, the **Undo** command description would have shown **Undo paste**.

Most times you would use **Undo** and **Redo** to back out or re-apply text editing operations. It is important to know that you can also use these features when changing tab stop settings and guides. Thus you can remove a tab stop you just created or move one back to where it was.

What Delimits an Undo and Redo Transaction?

The **Undo** command follows certain rules to define a transaction. It has to determine where a transaction starts and where it ends.

Most operations have a clearly defined scope. For example, if you select text and press the DELETE key, the whole selection is removed from the file. If you undo the operation, Qedit simply returns the deleted text. Some operations do not have as well defined limits. These are some of the rules that Qedit for Windows applies to undo and redo operations:

- A “typing” transaction starts with a printable character, a TAB, a SPACE, or a NEWLINE character. If there is no current selection, the BACKSPACE and DELETE keys also begin a typing session.
- Single-character deletions (using either the BACKSPACE or DELETE keys) that take place within a “typing” session are treated as if they were part of an original selection that was deleted.
- Exception to the above rules: any “typing” transaction that goes on for more than 24 lines gets made into its own transaction, and a new transaction is started.

Undo Command

Shortcut: CTRL+Z

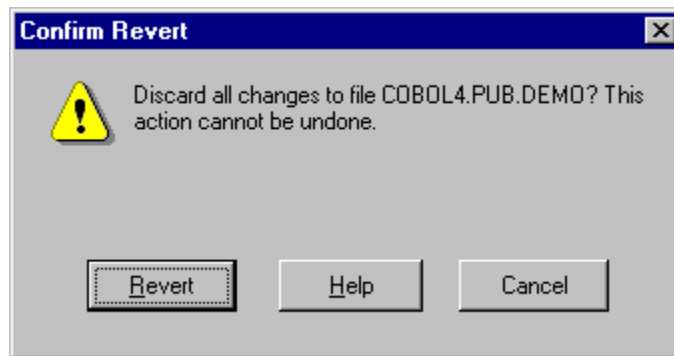
Toolbar: 

To undo the last modification, select the **Undo** command in the **Edit** menu. If you perform multiple undo actions, you cancel the next preceding update. You *cannot* undo changes out of order.

Undo All Modifications

The **Undo** function allows you to cancel modifications one at a time, starting with the most recent. Using the **Undo** command can be very tedious if you want to cancel all the modifications you have made. The **Revert** command in the **File** menu can do this much faster. This command cancels all the modifications made since you have either opened the file or last saved it, which ever came last.

NOTE: There is a major difference between the **Undo** and **Revert** commands. If you use **Undo** and you have gone too far back, you can always use the **Redo** command to re-apply changes. The **Revert** command *cannot* be reversed. For this reason, Qedit asks for confirmation before executing the **Revert** command.



Confirm Revert dialog box

Redo Command

Shortcut: CTRL+Y

Toolbar: 

If you want to re-apply a modification that you have just canceled, you can use the **Redo** command in the **Edit** menu. The **Redo** command is very handy if you are canceling consecutive modifications and go one step too far.

In standard Windows applications, such as Microsoft Word, “redo” means to reverse the last undo operation. It also has the same meaning in Qedit for Windows. Unfortunately, in Qedit for HP e3000 and Qedit for HP-UX, “redo” means to repeat a previous command. This confusion, however, is unavoidable in English.

Text-Formatting Features

Overview

Qedit for Windows has a number of practical text-formatting features that are useful in a programming environment. These features include various tools and methods for aligning text, as well as selecting fonts for documents and directories.

Fonts

Word processing applications, such as Wordpad and Windows Write, can highlight a heading and apply more than one font to a file. Qedit for Windows, however, does not offer multiple attributes in one file because host systems do not usually support them in their text files. To enable the transfer of programming and system files among disparate systems, we have sacrificed fancy typefaces.

Qedit is designed to deal with program source and data files. In these type of files, positioning text on specific columns is important. For this reason, Qedit allows you to choose only among fixed-pitch fonts installed on your PC.

Qedit for Windows allows you to select fonts that can be applied to both files and directory listings. It also lets you adjust the size and style of the fonts. These font attributes can be customized by each user. You can change the font of individual files or set the default font for all new files.

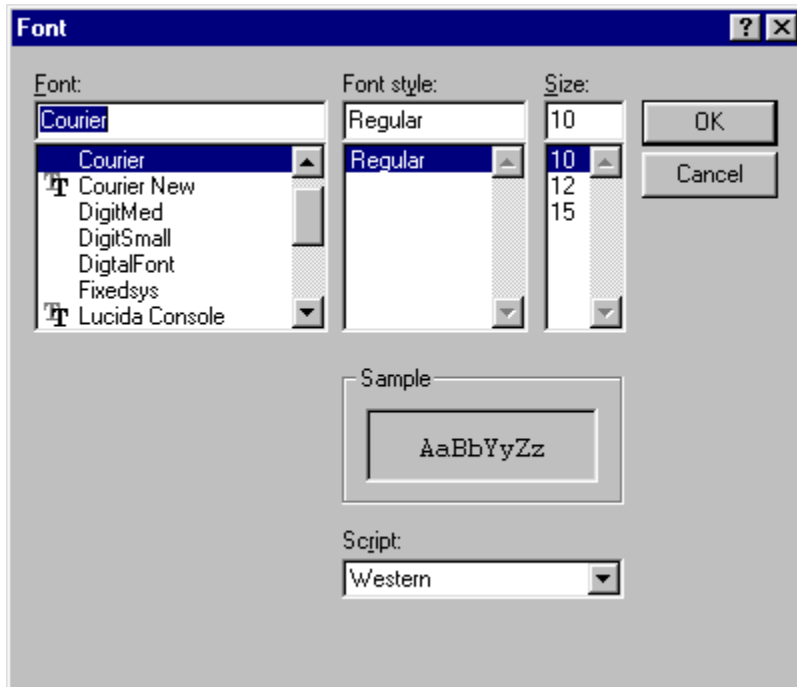
Changing the Document Font

NOTE: Qedit prints the font defined in the document (or the print driver uses the closest approximation available).

Qedit uses the font you have set as a default for document windows. Because it mostly works with data and program source files in which field position within a record is important, Qedit only displays fixed-pitch (monospaced) fonts. This way you know the column position on the screen is really the physical position in the record.

To change the font in a document,

1. Select the **Properties** command from the **File** menu.
2. When the **Properties** dialog box appears, click **Set font**.
3. When the **Font** dialog box appears, select the **Font**, the **Font style**, and the **Size**. If you want to apply a font size that is *not* available in the list, you can type a custom size in the text box.



Font selection dialog box

For information on setting the default font for files, see “Setting the Default Font for Document Windows” on page 121.

Changing the Directory Font

Qedit allows you to change the font in which the host directory list is displayed. To change the directory font,

1. Select **Preferences** from the **Options** menu.
2. When the **Qedit Preferences** dialog box appears, click **Directory Font**.
3. When the **Font** dialog box appears, select the **Font**, the **Font style**, and the **Size**. If you want to apply a font size that is *not* available in the list, you can type a custom size in the text box.

Bold, Italics and Underline

Qedit for Windows cannot apply **bold**, *italic*, or underline to a document because it is a programmer’s editor, not a word processor. Programming tools do not understand enhanced text with bold and fonts. They work only with plain ASCII files.

The directory listings in Qedit, however, can be set to bold, italic, and underline. These attributes can be changed in the **Font** dialog box. See “Changing the Directory Font” above.

Ruler Bar



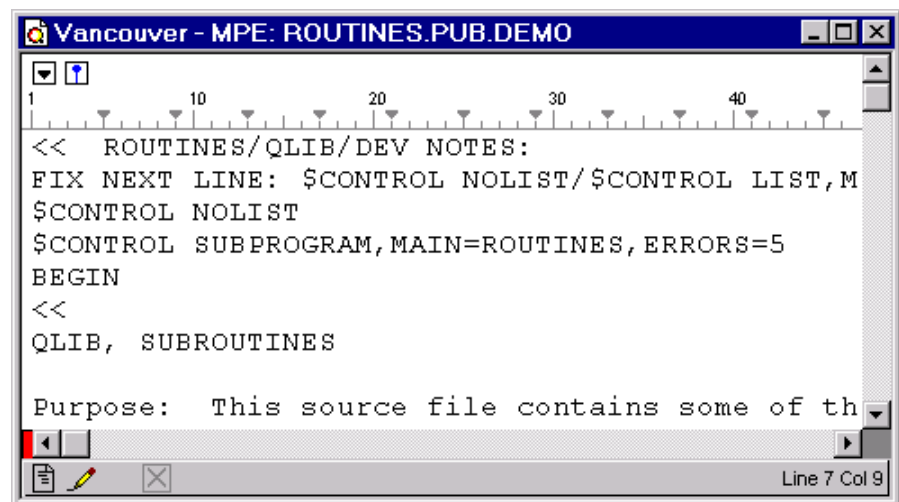
Tab stop tool



Guide line tool

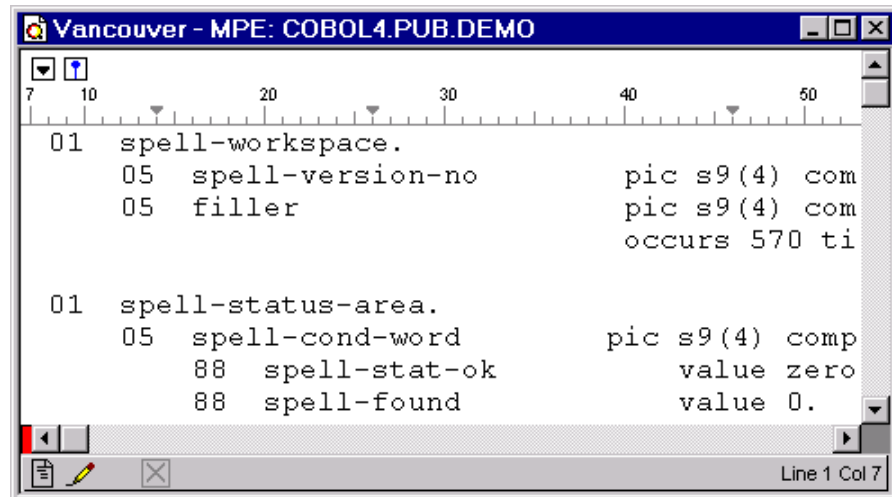
The ruler bar is a line at the top of a document window that is used to set tab stops and guide lines. Qedit for Windows can either show or hide the ruler bar. When it is enabled, the ruler bar as well as the tab stop and guide line markers appear at the top of the window.

The ruler bar has marks that indicate column positions. It shows numbers at regular intervals to make it easier to determine the cursor position. It also shows all the active tab stops and guide lines. For most files, the ruler starts in column one (1). The start column depends on the **Language** option selected.



Ruler with a text file

The ruler adjusts its column numbers according to the type of file you are editing. In the following example, when Qedit opens a COBOL source file, which starts in column seven (7), the first column on the ruler bar is number 7.



Ruler with a COBOL source file

Displaying and Hiding the Ruler Bar

Before you can set tab stops or guide lines, you have to show the ruler bar. To display the ruler bar in the file you are viewing, select **Ruler** from the **View** menu. To hide the ruler bar, simply repeat this step.

Tab Stops

Qedit allows you to set any number of tab stops in a document to help align the text in specific positions. If you press the TAB key, the cursor moves to the next tab stop relative to the cursor position. If there are no tab stops, a tab character only takes up one column, just any other character. You can set the tab stops either on the Qedit server or client.

Tab Stops on the Server

For complete syntax, refer to the user manual for host-based Qedit.

For host files, you can define tab stops right on the server. To do this, you need to add a Set Tabs command in one of the configuration files. When you open a new connection, the server passes the tab information to Qedit. The new tab stops are then used for all files opened on that connection. For example, to set tab stops on columns 8, 12, 16 and 24, you would have the following lines in one of the Qeditmgr files:

```
if Qedservmode <> 0 then
  /set tabs 8 12 16 24
endif
```

You have to be careful where you place the Set Tabs command in the configuration file. The tab values are relative to the first column of the file, which is defined by the **Language** option. Because different language settings have a different starting column, the result might not be what you expect. For example, if you have the following in your configuration file:

```
if Qedservmode <> 0 then
  /set Language SPL
```

```
    /set tabs 8 12 16 24
endif
```

the tab stops will be set at columns 8, 12, 16 and 24 because the first text column in an SPL file is one (1). However, if you have

```
if Qedservmode <> 0 then
  /set Language Cobol
  /set tabs 8 12 16 24
endif
```

the tab stops would be set at columns 14, 18, 22 and 30 because the first text column in a COBOL file is 7.

Tab Stops on the Client

When you change tab stops on the client, the new settings are recorded in the document database and are used whenever you access the file. Tabs modified on the client are only applied to the active file.

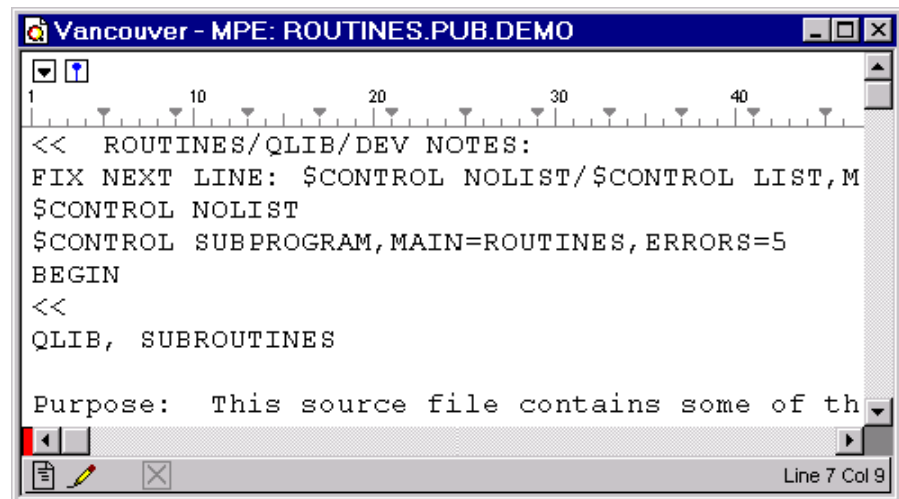
There are two ways you can set document-specific tab stops on the client: on the ruler of the document window or in the **Properties** dialog box.

Setting Tab Stops on the Ruler

Qedit shows tab stops on the ruler bar as small arrowheads pointing downward. If you have never modified the tab stops (e.g., new file or file never modified), Qedit's default is every four columns. New tabs can be set with a tab stop marker, which appears in a button above the ruler.



Tab stop marker



Default tab stops

Before you can set a tab stop, the ruler bar must be visible. To add new tab stops, drag the marker to a position on the ruler. Click and hold the marker until the mouse pointer becomes an arrowhead. Then drag the tab marker to a position on the ruler bar. Tab stops apply to the whole file.

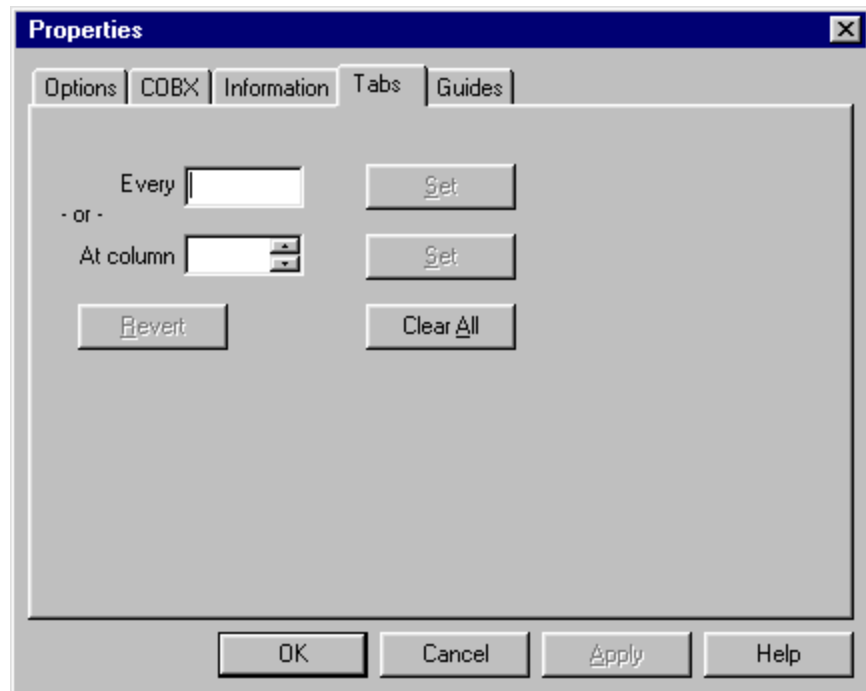
Moving or Clearing Tab Stops on the Ruler

Before you can move or clear a tab stop, the ruler bar must be visible. To move a tab stop, drag the marker to the new position on the ruler bar. To remove a tab stop, drag it off the ruler bar.

NOTE: If the **Use spaces for tabs** document option is enabled, changing tab stops does not affect existing lines. Because the text does not contain TAB characters anymore, the alignment remains the same. However, if the document option is disabled and there are TAB characters in the text, the text will shift as you modify the tab stops.

Setting Tab Stops in the Properties Dialog Box

The other method for setting tab stops is through the **Properties** dialog box on the **Tabs** tab. You can easily set an individual tab stop or a series of tab stops.



Setting tab stops in the Properties dialog box

To set individual tab stops,

1. Select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, select **Tabs**.
3. Enter a column number in the **At column** text box. You can type a number or you can use the scroll arrows to select the column you want.
4. If the new tab stop does not overlap with any current tab stops, the button next to the **At column** text box will be called **Set**. Click it to acknowledge the new tab.

If the new tab overlaps with any current tab stops, the button next to the **At column** text box will be called **Clear**. To remove the tab that overlaps, click **Clear**. To allow overlapping tabs, click the **Clear** button to change it to **Set**, then click the button again.

5. For each new tab stop, repeat steps 3 and 4.
6. To save the new tab(s) and close the dialog box, click **OK**. To set the new tab(s) without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

To set tab stops at regular intervals,

1. Select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, select **Tabs**.
3. Enter the number of columns between each stop in the **Every** text box. Remember that these intervals start at column 1. For example, if you specify a stop every 8 columns, tab stops will be set at columns 9, 17, 25, etc.
4. If the new tab stops do not overlap with any current tab stops, the button next to the **Every** text box will be called **Set**. Click it to acknowledge the new tab stops.

If the new tab stops overlap with any current tab stops, the button next to the **Every** text box will be called **Clear**. To avoid setting tabs in overlapping positions, click **Clear**. To allow overlapping tabs, click the **Clear** button to change it to **Set**, then click the button again.

5. To save the new tabs and close the dialog box, click **OK**. To set the new tabs without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

If you have the ruler bar enabled, the markers for these new tab stops are automatically displayed in the document window.

Moving Tab Stops in the Properties Dialog Box

To move a tab stop in the **Properties** dialog box, you need to add a tab at each new location (see “Setting Tab Stops in the Properties Dialog Box” above) and clear each old tab stop (see “Clearing Tab Stops in the Properties Dialog Box” below).

Clearing Tab Stops in the Properties Dialog Box

You can clear overlapping tab stops when you set new ones (see “Setting Tab Stops in the Properties Dialog Box” above). After tabs have been set, however, you can also clear one tab, a series of tabs at regular intervals, or all the tab stops in the file.

To clear some of the tab stops,

1. Select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, select **Tabs**.
3. Enter the position of the tab(s) that you want to remove in either the **At column** (individual tab) or **Every** (series of tabs) text box. If you have entered the position of an existing tab, click the **Clear** button next to the text box. If you have entered a position that *does not* correspond to an existing tab, the button will be called **Set**.

Repeat this step for each tab or set of tabs that you want to remove.

4. To clear the selected tabs and close the dialog box, click **OK**. To clear the selected tabs without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

If you want to remove all the tab stops from a file, simply select **Properties** from the **File** menu. When the **Properties** dialog box appears, select **Tabs** and click **Clear All**.

Auto-Indent

Auto-indent is a great feature because it allows you to follow the indent of the last line when you press the RETURN or ENTER key. Instead of the cursor going back to the first position of the line, it goes to the column that matches the indentation of the previous line. You can use spaces or tabs to increase the indentation. If you use tabs, you can use SHIFT+TAB to decrease the indentation.

You can change the auto-indent setting for a specific document with:

1. Select the **Properties** command from the **File** menu.
2. When the **Properties** dialog box appears, go to the **Options** tab.
3. Select the **Auto-indent** check box.

Guide Lines

Guide lines are vertical lines that provide a visual aid for aligning text. They do not modify your text, but provide you with a constant reference point throughout a file. Once you set a guide line in a document, it remains on your screen as long as you view the file.

There are two ways you can set document-specific guides: on the ruler of the document window or in the **Properties** dialog box.

Setting Guide Lines on the Ruler

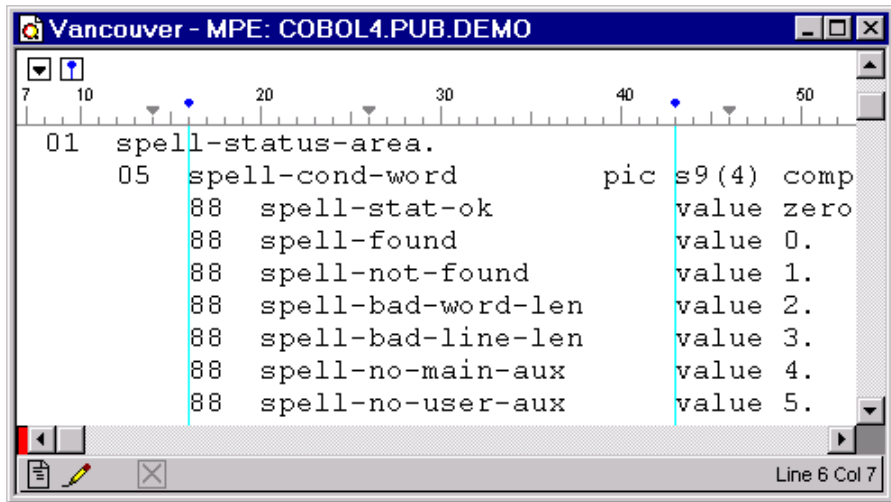
Before you can set a guide line, the ruler bar must be visible. To set a new guide line, click and hold the **Guide** marker above the ruler bar until the mouse pointer changes to a **Guide** pointer (a square pinhead). Then drag this new pointer to a position on the ruler bar and release it. The new guide line appears as a thin blue vertical line from the ruler bar all the way down to the document status bar. You can set as many guide lines as you need.



Guide marker



Guide pointer



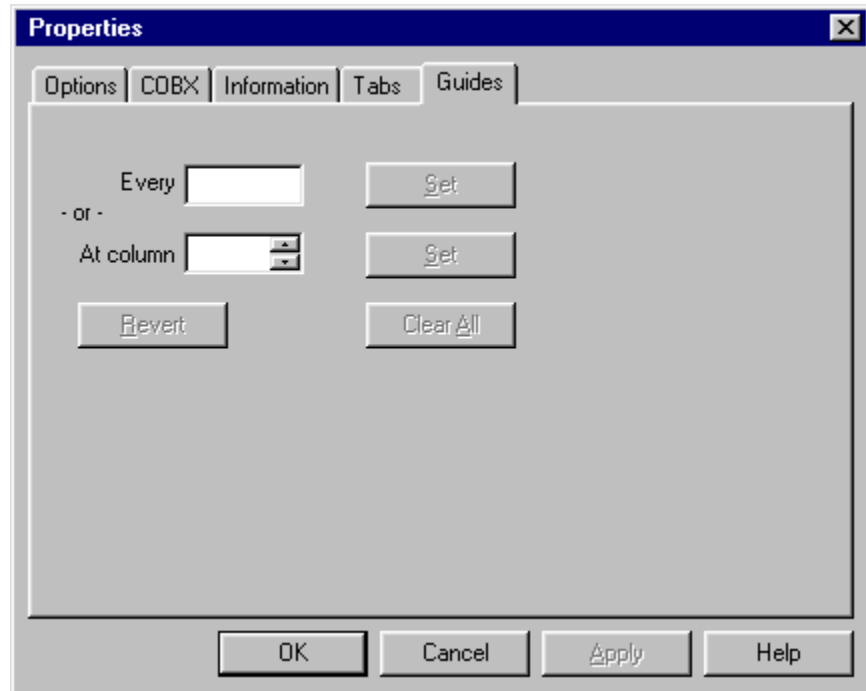
Guide lines in document window

Moving or Clearing Guide Lines on the Ruler

If you want to move an existing guide line, simply drag it to a new position on the ruler bar. If you want to remove a guide line, drag it off the ruler bar.

Setting Guide Lines in the Properties Dialog Box

The other method for setting guide lines is through **Properties** dialog box on the **Guides** tab. You can easily set an individual guide line or a series of guides.



Setting guide lines in the Properties dialog box

To set individual guides,

1. Select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, select **Guides**.
3. Enter a column number in the **At column** text box. You can type a number or you can use the scroll arrows to select the column you want.
4. If the new guide line does not overlap with any current guide lines, the button next to the **At column** text box will be called **Set**. Click it to acknowledge the new setting.

If the new guide line overlaps with any current guide lines, the button next to the **At column** text box will be called **Clear**. To remove the overlapping line, click **Clear**. To allow an overlapping line, click the **Clear** button to change it to **Set**, then click the button again.

5. For each new guide line, repeat steps 3 and 4.
6. To save the new guide line(s) and close the dialog box, click **OK**. To save the new guide line(s) without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

To set guides at regular intervals,

1. Select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, select **Guides**.
3. Enter the number of columns between each stop in the **Every** text box. Remember that these intervals start at column 1. For example, if you specify a stop every 8 columns, guide lines will be set at columns 9, 17, 25, etc.

4. If the new guide lines do not overlap with any current guide lines, the button next to the **Every** text box will be called **Set**. Click it to acknowledge the new setting.

If the new guide lines overlap with any current guide lines, the button next to the **Every** text box will be called **Clear**. To remove only the guide lines that overlap, click **Clear**. To allow overlapping guide lines, click the **Clear** button to change it to **Set**, then click the button again.

5. To save the new guide lines and close the dialog box, click **OK**. To set the new guide lines without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

If you have the ruler bar enabled, the new guide lines are automatically displayed in the document window.

Moving Guide Lines in the Properties Dialog Box

To move guide lines in the **Properties** dialog box, you need to add a guide line at each new location (see “Setting Guide Lines in the Properties Dialog Box” above) and clear each old guide lines (see “Clearing Guide Lines in the Properties Dialog Box” below).

Clearing Guide Lines in the Properties Dialog Box

You can clear overlapping guide lines when you set new ones (see “Setting Guide Lines in the Properties Dialog Box” above). After guide lines have been set, however, you can also clear one line, a series of lines at regular intervals, or all the guide lines in the file.

To clear some of the guide lines,

1. Select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, select **Guides**.
3. Enter the position of the guide line(s) that you want to remove in either the **At column** (individual guide line) or **Every** (series of guide lines) text box. Once you have entered the position of an existing guide line, click the **Clear** button next to the text box. If you have entered a position that *does not* correspond to an existing guide line, the button will be called **Set**.

Repeat this step for each guide line or set of guide lines that you want to remove.

4. To clear the selected guide line(s) and close the dialog box, click **OK**. To clear the selected guide lines without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

If you want to remove all the guide lines from a file, simply select **Properties** from the **File** menu. When the **Properties** dialog box appears, select **Guides** and click **Clear All**.

Invisible Formatting Characters

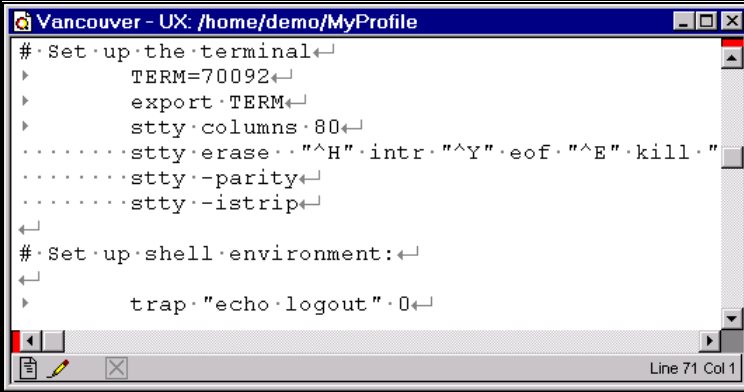
Some characters, such as spaces and tabs, are used to format text in a file. By default, they do not appear on the screen when you edit a file; they appear as empty

space between characters. Even though they are mostly used for lining up columns in a file, they can also have special meanings. For example, files on UNIX can contain TAB characters to separate data fields.

Qedit also displays end-of-line markers whenever appropriate.

Shortcut: Alt+V, I

If you want to see where SPACE and TAB characters are used, you can enable the **Show invisibles** command on the **View** menu. Once enabled, spaces between words are shown as small dots. Each dot represents one character. TAB characters are represented by arrowheads pointing to the right. End-of-line markers are represented by a left-pointing arrow. This icon is similar to the one usually appearing on the keyboard ENTER key.



```
Vancouver - UX: /home/demo/MyProfile
# Set up the terminal
> TERM=70092
> export TERM
> stty columns 80
.....stty erase ^H intr ^Y eof ^E kill .
.....stty -parity
.....stty -istrip
←
# Set up shell environment:
←
> trap "echo logout" 0
Line 71 Col 1
```

Invisible formatting characters displayed

In the example, you can see the lines in the “Set up the terminal” section are aligned. The first three use TAB characters and the last three use SPACE characters.

The color in which these invisible formatting characters appear is defined in the color scheme applicable to the document.

Matching Opening and Closing Characters

In most programming languages, syntax requires the use of opening and closing characters to delimit operations or expressions. Most common character pair is probably parentheses. Another example is the left and right square brackets "[" and "]" in Qedit Scripting Language. To complicate things further, often times, the character pairs can be nested making it hard to find matching sets. Qedit offers a simple feature in which the opening character is momentarily highlighted as you type a matching closing character. This feature helps identify the text enclosed by the pairs. The opening character is briefly highlighted with a square using the **Attention highlight** color defined in the color scheme for this file.

This feature can be disabled by removing the check from the **Balance Punctuation (parentheses, brackets) while typing** option on the **General** page of the **Preferences** dialog box.

Also, if you type a closing character for which Qedit cannot find a matching opening character, Qedit emits a sound. You can disable this feature by removing the check from the **Alert sound on unmatched punctuation** option on the **General** page of the **Preferences** dialog box.

This feature recognizes parentheses, square brackets and curly braces.

Working With Syntax Coloring

Syntax coloring uses different colors to identify different parts of the text. This is useful for any text that follows syntactic rules such as programming languages. In these structured text, words are not created equals and have different usage. In Qedit for Windows, there are 4 categories:

- regular text
- comments
- delimited or quoted text
- keywords or reserved words

Regular text typically represents user-defined words such as program variables. Comments are sections providing hints and documentation. They usually don't have any effect on the programs using the file. Most programs need to manipulate character strings. Typically, these strings are enclosed between delimiters or quotes. Lastly, certain words have special meanings. They are usually known as reserved words or keywords.

Qedit uses language template files to describe language syntax elements. A template file describes how to recognize the language of a file, comments, quoted text and what are the keywords. Each element is assigned a color. During an edit session, Qedit analyzes the text and highlights words with the appropriate color. This event occurs automatically as soon as Qedit is able to associate a language. For new files, syntax coloring might occur only after the first save. For existing files, this should happen as soon as the file is opened. If it can't associate a language, no coloring occurs.

Language template files are loaded when Qedit starts up. Since these files are plain text files based on the XML specifications, you can easily customize existing templates or you can create your own. To activate your changes, you need to exit out of Qedit and start it up again.

If you wish to know which language Qedit is associated with a particular file, look at the **Information** page of the **File Properties** dialog box. The language name on the page is the name specified in the language template file.

Ambiguous Language Identification

Qedit associates a language with a file based on one of 2 things:

- a Qedit Language Code for host files
- the extension of a file for Posix, Unix and local files.

This information is entered in each language template. Qedit does not prevent the same language code or extension from being used in more than one template. Qedit uses a very simple method to resolve the ambiguity: it uses the last one loaded. From the user's point of view, however, this can be disconcerting. We recommend that a language code or file extension be used in only one language template.

SQL Language Templates

The Structured Query Language (SQL) is used with most, if not all, relational database management systems (RDBMS) such as Oracle, Allbase and MySQL. Although these systems are based on SQL, they all have differences and unique

language extensions. For this reason, Qedit for Windows comes with multiple SQL template files: oracle.xml, allbase.xml and mysql.xml. Most of these systems allow creation and execution of script files. Typically, the script files share the same extension: ".sql". That's what is used in the template files.

To avoid any confusion and make sure Qedit picks the appropriate template for your environment, you have to remove the ambiguity. To do this, you can do one of the following:

- You can rename the templates you do not use. Change the .xml extension to something else.
- You can move the files from \robelle\qedit\system\language to another directory.
- You can edit the templates you do not intend to use. Open them in Qedit and remove the **extension** blocks or change the extension identifier to something unique.

Order of Precedence

The same characters or strings can be used in different categories in a language template. This can affect coloring behavior and may appear incorrect to users. That's why this is not recommended.

When analyzing text for syntax coloring, Qedit uses the following order of precedence to identify categories:

1. Quotes
2. Comments
3. Keywords
4. Regular text

For example, if double-quote character is defined as a quote delimiter as well as a comment delimiter, it's going to be used as a quote. Similarly, if the letter "C" appears as a comment delimiter as well as in the keyword list, it's going to be used as the start of a comment.

Text within quotes is not analyzed. Similarly, text following the start of a comment is not analyzed.

Customizing Text-Formatting Features

Displaying the Ruler Bar When Opening Files

The ruler bar can be enabled in each document. If you would like every document window to open with the ruler bar, first select **Preferences** from the **Options** menu. Then enable the **Open file with a ruler bar** option from the **Default** tab.

NOTE: If this option is enabled, the ruler bar will appear in all the panes created with the split view feature.

Setting the Default Font for Document Windows

If you would like all your files to have a specific font, you set it as Qedit's default. To set the default font for all new documents,

1. Select **Preferences** from the **Options** menu.
2. When the **Qedit Preferences** dialog box appears, go to the **Defaults** tab and click **Display Font**.
3. When the **Font** dialog box appears, select the **Font**, the **Font style**, and the **Size**. If you want to apply a font size that is *not* available in the list, you can type a custom size in the text box.

Once the default font is changed, any new document you open will automatically use the new font. Any open documents at the time of the change will be converted to the new font. However, if you change the default font of a file with the **Properties** command in the **Files** menu, the font in open documents will not change.

Setting the Default Value for Auto-indent

If you would like to use the same auto-indent setting for all files you have never opened, you can set it in the **Preferences** dialog box:

1. Select **Preferences** from the **Options** menu.
2. When the **Qedit Preferences** dialog box appears, go to the **Defaults** tab and set **Enable auto-indentation** to the desired state.

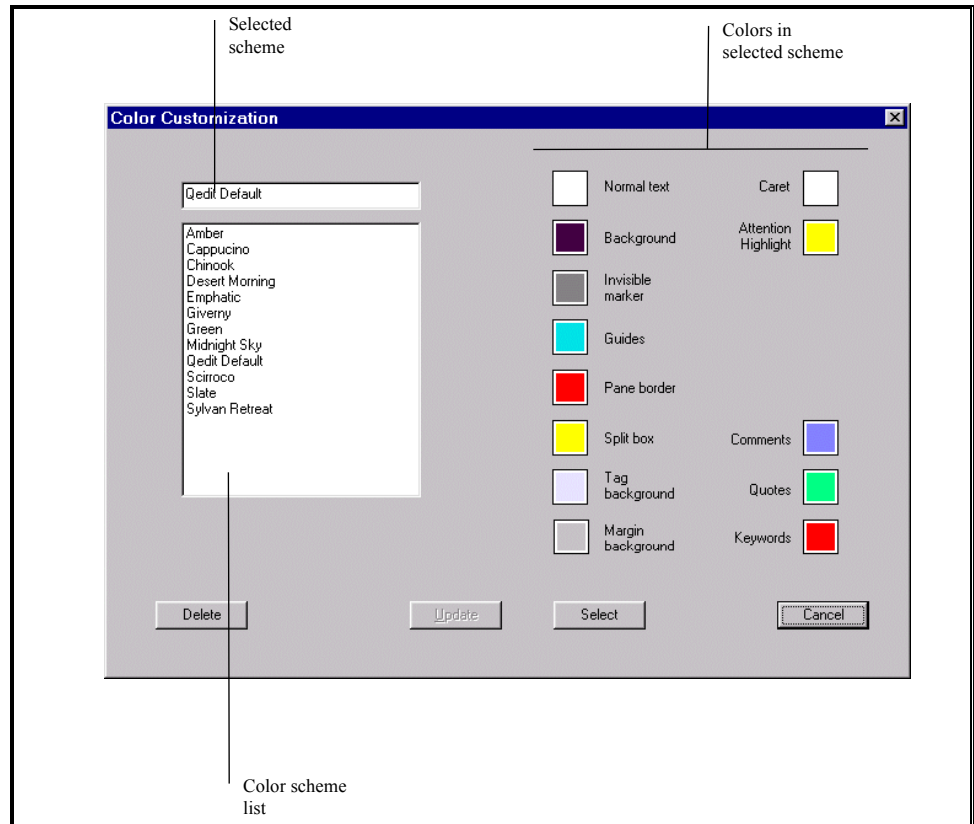
Any new document you open will automatically use the new setting.

Displaying Invisible Formatting Characters When Opening Files

To display invisible formatting characters, enable the **Show invisibles** option in the **View** menu. If you prefer to have invisible formatting characters always appear when opening a file, enable the **Open files with invisibles shown** option on the **Defaults** tab of the **Preferences** dialog box.

Setting the Default Color Scheme

Qedit allows you to define your own color schemes for document windows. It offers a number of predefined schemes, including one for Qedit defaults. You can select a color scheme as the overall default from the **General** page of the **Preferences** dialog box.



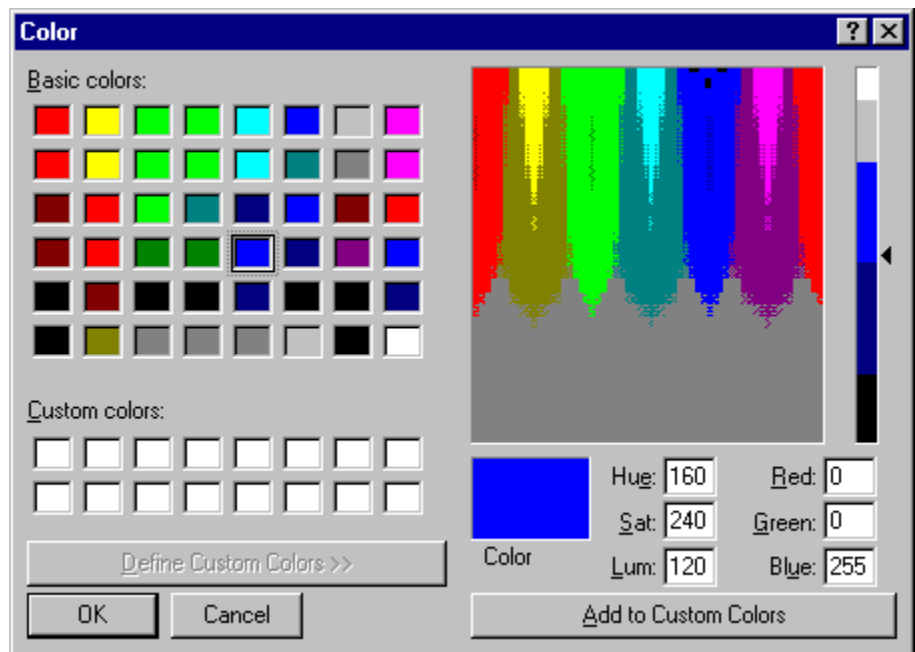
Color Customization dialog box

To select a color scheme, simply click the name in the color scheme list. Its colors appear in the color boxes on the right. Each color scheme includes these elements:

- **Normal text:** Characters making up the actual text of the file and COBOL tags.
- **Background:** Workspace background. Characters in the foreground can be **Normal text**, **Invisible marker**, **Guides**, **Comments**, **Quotes** and **Keywords**.
- **Invisible marker:** Markers displayed when the **Show invisibles** option of the **View** menu is enabled.
- **Guides:** Once set, guidelines are always visible in the document window. Guidelines can be set when the **Ruler bar** is displayed or from the **Guides** page of the **Properties** dialog box.
- **Pane border:** Panes appear when using the split view feature. This color is used to delimit the panes.
- **Split box:** Small box at top of the vertical scroll bar or on the left of the horizontal scroll bar. The box is used to create split views.
- **Tag background:** Area where COBOL tags are displayed. The tag values can be displayed using one of the **Normal text**, **Comments**, **Quotes** or **Keywords** colors.
- **Margin background:** Area between the right margin of the text and the right margin of the document window. This area should be empty.

- **Caret:** Blinking insertion point (vertical or horizontal) when there is no text selected.
- **Attention highlight:** Brief visual signal to get the user's attention. For example, the corresponding opening parenthesis when typing a closing parenthesis.
- **Comments:** Various types of comments defined for the file's language (see "Appendix D - Language Templates" on page 251).
- **Quotes:** Text enclosed in quote characters defined for the file's language (see "Appendix D - Language Templates" on page 251).
- **Keywords:** Language reserved words (see "Appendix D - Language Templates" on page 251).

To select a new color, click the corresponding color box. You will see another dialog box with a color palette:



Color palette dialog box

Select the color that you want and click **OK**. The color palette shown above could be different on your system, because color is based on your current display settings.

Here is how a sample Cobol source program is displayed with syntax coloring.

Sample Cobol source file with syntax coloring.

Changing an Existing Color Scheme

To change an existing color scheme,

1. Select **Preferences** from the **Options** menu.
2. Select **Color schemes** on the **General** page.
3. Select a color scheme in the list or type its name.
4. Select the color box of the corresponding element.
5. Select the new color from the color palette.
6. Click **OK**.
7. Click **Update** on the **Color customization** dialog box.
8. Click **Cancel** to close the dialog box.

Adding a Color Scheme

To create a new color scheme,

1. Select **Preferences** from the **Options** menu.
2. Select **Color schemes** on the **General** page.
3. If you want to use an existing scheme as the base, select it from the list.
4. Type in the new name (replace the existing name if needed). Color scheme names can only contain letters, numeric digits, spaces, underscores, asterisks, pound signs, exclamation point, dots, slashes and commas.
5. Select the color box of the element you want to change.
6. Select the new color from the color palette.
7. Click **OK**.
8. Change the necessary elements.

9. If the name did not exist, the **Update** button on the **Color customization** dialog box becomes **Add**. Click it to finalize the new scheme.
10. Click **Cancel** to close the dialog box.

Removing a Color Scheme

To remove an existing color scheme,

1. Select **Preferences** from the **Options** menu.
2. Select **Color schemes** on the **General** page.
3. Select the color scheme from the list or type it.
4. Click **Delete** to remove it.
5. Click **Cancel** to close the dialog box.

Default Colors Hierarchy

Default colors can be set at different levels.

1. **Global default:** Defined from the **General** page of the **Preferences** dialog box.
2. **Host type defaults:** Defined from the **UNIX**, **MPE** and **Local** pages of the **Preferences** dialog box.
3. **Connection default:** Defined from the **Connection List** dialog box.
4. **File colors:** Defined from the **Options** page of the **Properties** dialog box.

Colors defined at lower levels always override colors defined at previous levels. For example, if you have defined host type defaults, these will be used even if a global default has been set. Similarly, file colors are always used, no matter what has been defined for the corresponding connection, host type or globally.

Line Features

Overview

Qedit's many line features can be applied to an array of situations. Using line numbers and record numbers, you can quickly navigate through files. Display COBOL tags when you need to see them, define the maximum line length of a file, and control spaces at the end of a line. All these options expand the Qedit toolkit.

Record Numbers

Lines can be referenced by either a number relative to the beginning to a file (record number) or a number assigned by a host (line number). Qedit for Windows can use these two types of numbers as navigational tools.

Host Line Numbers

Some Qedit for Windows host files have line numbers, which are sequential numbers assigned by the host. Although line numbers are stored within a host file, they do not appear in the document window. Line numbers can have up to 3 decimals e.g. 115.832. See "Finding a Line Number in a Host File" on page 141 for more information.

Record Numbers

All files in Qedit for Windows have record numbers, which are ordinal numbers that represent a line position relative to the beginning of the file. Thus, record numbers can be integers only, no decimals. When this option is enabled, record numbers appear in the document window. See "Finding a Record Number" on page 142 for more information.

Displaying Record Numbers

If the **Record number** command is enabled on the **View** menu, Qedit shifts the text to the right and inserts the record number of each line. These numbers appear dimmed because they cannot be modified.

```
13 Install: :restore *qtape;ROUTINES.QLI
14
15 Doc: See CONTENTS.QLIBDOC.
16
17 Contents:
18
19 COPYCOL - copy columns in qedit fil
20 NEATER - Qedit /P routine to align
21 QOUT - Subroutines to write QEDI
22 FINDJUNK - Editor and QEDIT /P routi
23 >>
```

Record numbers in document window

Hiding Record Numbers

If you do not want line numbers to appear in the document window, you can hide them. To hide line numbers in a file, select **Record numbers** in the **View** menu.

Trailing Blanks

Preserve Trailing Blanks

By default, Qedit always preserves trailing spaces that have been entered. If you try to insert new text on these lines, the trailing spaces will eventually reach the right margin. When this happens, Qedit will not allow you to enter anything else. You then have to go to the end of the line and remove some of these spaces to make room for the extra characters. Although this can be annoying, Qedit does this because it presumes that anything you type explicitly is valuable and should be kept intact.

Unfortunately, you cannot tell which lines have trailing spaces and which ones don't. If trailing spaces are really not important, you can disable **Preserve trailing blanks**. Once disabled, Qedit removes trailing spaces as needed to make room for new text.

This setting only affects editing while the file is opened. As you modify the file and add trailing blanks, Qedit recognizes them. However, trailing blanks are not written to the file when you save it. Next time you open the file, trailing blanks are gone. That's because the server never transmits them to the Qedit client.

NOTE: If you delete characters from a line, Qedit does not put back trailing spaces that might have been removed.

COBOL Tags

Tags can only be used in files with Language of COBX.

COBOL source files can contain some information in columns 73 to 80 of each line. This area is sometimes referred to as auto-comments or tags. For COBOLX files, this area appears in a different color. The color is defined in the color scheme applicable at the time. You can set COBOL tags in either the server configuration file or the client.

COBOL Tags in Server Configuration Files

Although the default value for a COBOL tag is set in the server configuration file, you can override it by using the **COBX** tab in the **Properties** dialog box.

For example, if you want to set a special default tag only for users that connect as “CLERK” in their session name, you can either insert the following lines in Qeditmgr.Pub.Robelle or each user can put the Set X command in their own Qeditmgr configuration file. See “Customizing the Server” on page 19 for more details.

```
if hpjobname="CLERK" then
  Set X "ME" yymmdd List on
endif
```

The Set X command has two parts. The first part, Set X *<string>* *<date>*, allows you to define the string you want inserted in the tag area. The second part, Set X List On | Off, allows you to control the display of the tag area. In total, you have four (4) possible combinations:

- Set X "" List Off
No automatic or manual update, tags are hidden
- Set X "" List On
Manual update allowed, no automatic update, tags are displayed
- Set X *<string>* *<date>* List Off
Automatic update, no manual update allowed, tags are hidden
- Set X *<string>* *<date>* List On
Automatic update, no manual update allowed, tags are displayed

Set X List On vs. Set X List Off

The List option of the Set X command controls whether or not you see the tags in the document window. The text area only goes to column 72. Because you do not have access to columns 73 to 80, you cannot manually enter tags.

```

01 spell-status-area.
05 spell-cond-word      pic s9(4) comp.
88 spell-stat-ok       value zeros.
88 spell-found         value 0.
88 spell-not-found     value 1.
88 spell-bad-word-len  value 2.
88 spell-bad-line-len  value 3.
88 spell-no-main-aux   value 4.
88 spell-no-user-aux   value 5.
88 spell-no-both-aux   value 6.

```

COBOL file with tag area not displayed

If this option is enabled, the area in columns 73 to 80 appears in a different background color. All existing tags are displayed in that area. The text area and the tag area are independent of each other. If you delete characters from the text, the tag value does not move to the text area. Similarly, if you insert text, characters from the text area do not overflow into the tag area. If you split a line, the tag value remains on the original line. If you splice a line by deleting the end-of-line character, the tag value from the second line does not move with the text.

```

05 filler      pic s9(4) comp
               occurs 570 times.
*
* New comment lines
*
01 spell-status-area.
05 spell-cond-word      pic s9(4) comp.
88 spell-stat-ok       value zeros.
88 spell-found         value 0.
88 spell-not-found     value 1.

```

COBOL file with tag area displayed

Set X vs. Set X <string> <date>

The other parameter of the Set X command allows you to update the tags either manually or automatically. If you do not want Qedit to automatically tag modified lines and would prefer to manually enter something in the tag area, put a

```
set x
```

command with no parameters in the configuration file or use

```
set x ""
```

Of course, this requires Set X List On to have the tag area displayed.

(a null string can be entered as two consecutive quotes). The tag area is then under your total control. Qedit will not automatically put or change anything in these columns. With this setting, you can move the insertion point to the tag area and start typing. If the cursor is in the tag area, Qedit automatically switches to overwrite mode.

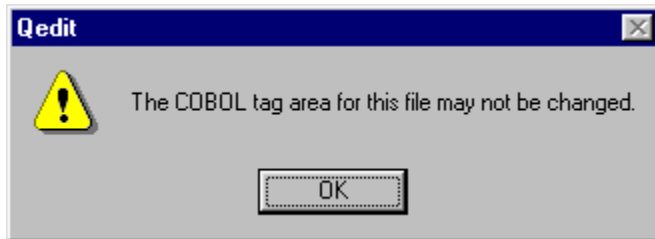
On the other hand, if you want Qedit to put a tag on all modified and new lines, use

```
set x <string> <date>
```

There are many different ways to define the tag value. You can use a character string, the host system date in various formats or combine a character string with the system date. Please refer to the Set X command in the host-based Qedit user manual for details.

If you have defined a tag value, Qedit automatically inserts it on modified lines *whether you see the tag area or not*. In the example used previously, the tag would contain the letters “ME” followed by the host system date in YYMMDD format. For example, on December 31, 1977, the value would have been “ME771231.”

The tag area is then under Qedit’s total control and you will not be able to manually change anything in it. If you try to modify a tag, you will get an error message:



COBOL tag modification error

COBOL Tags in the Client

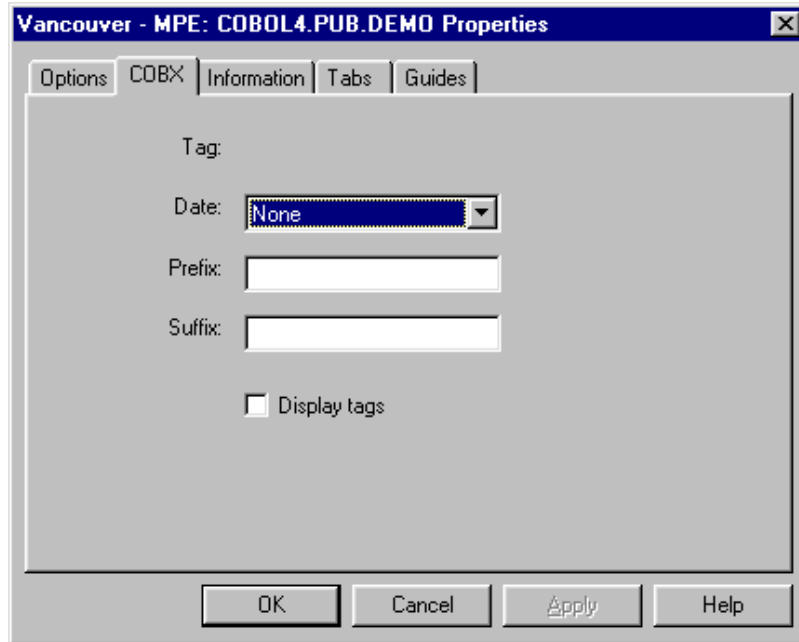
If you want to customize your COBOL file, you can also change the settings of the COBOL tags in the client. These settings override the tag settings in the server configuration files.

Setting COBOL Tags in the Client

Changes to tag settings in the client are applied only to the current file and are saved in the document database. New tag values are also sent to the server so that modified lines are properly identified.

To set COBOL tags in the client,

1. Select **Properties** in the **File** menu.
2. When the **Properties** dialog box appears, click **COBX**. Although you can always access this tab, you will not be allowed to change any of its settings unless:
 - the **Language** of the file is **COBX**
 - you are running version 4.7.11 and later of the server



Changing the COBOL tag in the COBOL tab

3. Enter the new COBOL tag settings in the **Date**, **Prefix** and **Suffix** boxes. A COBOL tag can have a literal prefix (string), a literal suffix (string), and the system date. You can use only one of these settings or combine any of them. Qedit has only one restriction: the total length of the tag cannot exceed 8 characters. As you enter new settings, Qedit displays the new tag beside the **Tag** label.

There are six different formats in the **Date** list box:

- DDMmmYY, where the month (Mmm) appears as a 3-letter abbreviation where the first letter is uppercase and the last 2 are lowercase.
- YYMMDD
- MMDDYY
- DDMMYY
- CCYYMMDD
- DD MMMYY, where the month (MMM) appears as a 3-letter abbreviation, all uppercase.

In the formats with the 2-character month (MM), the information is all numeric: CC represents the century, YY represents the year, MM represents the month and DD represents the date.

4. To control whether Qedit will show or hide the new tag in your document window, simply enable or disable the **Display tags** option.

This option only controls the display. It does not affect the way lines are updated. Basically, if a tag value is defined in **Date**, **Prefix** or **Suffix**, modified lines are tagged even if you don't see the results on the screen.

5. To save the new settings and close the dialog box, click **OK**. To save the new settings without closing the dialog box, click **Apply**. The **Apply** button is useful when you want to continue working in the **Properties** dialog box.

If the new tag exceeds the maximum length (8 characters), Qedit will display an error message when you select **Apply** or **OK**.

COBOL Tags: Automatic vs Manual

COBOL tags can be updated automatically or entered manually but these options are mutually exclusive. The rules are fairly simple. If you have defined a tag value, Qedit assumes you want to use the automatic update feature. Whether tags are displayed or hidden does not have any effect.

If you have not defined a tag value, Qedit assumes you want to be able to manually modify lines. Of course, the **Display tags** option must be enabled so you can have access to the tag area.

Line Length

Maximum Line Length

The maximum line length can be as much as 8,172 characters or as small as 66 columns for COBOL, or even smaller for data files. This means that when you combine lines together by pressing DELETE at the end of a line or by pasting from the Clipboard, you will likely receive the message:

```
The resulting line would be too long
```

Most Windows editors have such a large maximum line length, such as 64,000 columns, that you never see this problem. But many host-based files *must* have fixed-size lines, both programming language files and application data files.

Line Termination

Lines in UNIX files are usually delimited with the NEWLINE character. It is sometimes called the LINE FEED character. Other operating systems use a CARRIAGE RETURN or a combination of CARRIAGE RETURN and LINE FEED characters. Qedit automatically detects these characters and splits lines at the correct location. During a **Save** operation, Qedit uses the same record delimiter.

You can change the line termination character on the **Options** page of the **Properties** dialog box or the **Local** page of the **Preferences** dialog box.

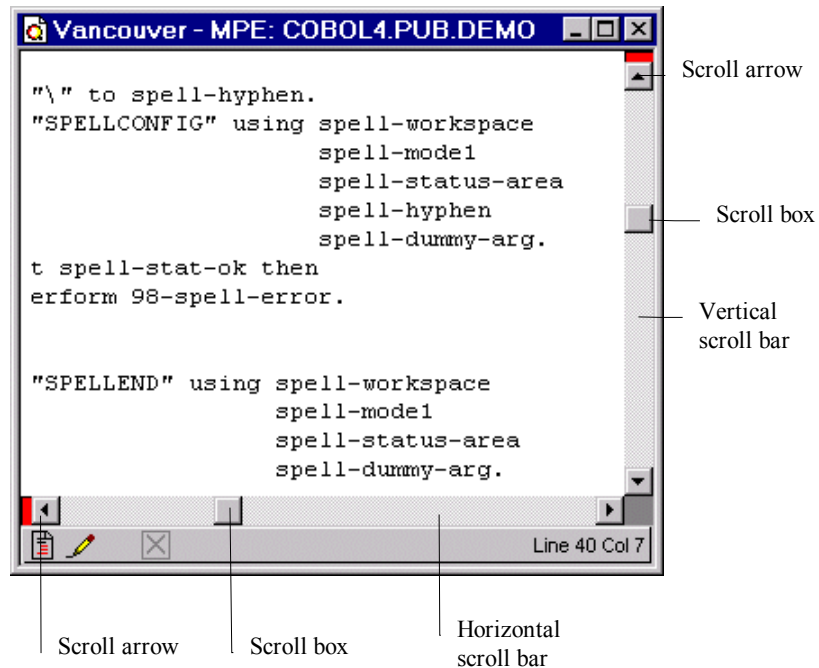
Searching & Replacing Features

Overview

Qedit gives you immediate search results, even with enormous host files. This is because Qedit actually performs host file searches on the server, not the client. When Qedit finds a string match, it downloads only the portions of the file that contain the match. That's why Qedit is so fast. Qedit also provides you with numerous search criteria: embedded words, highlighted words, pattern matching, as well as a partial file search. Qedit's replace option further enhances the search capabilities.

Scrolling Through a File

You can scroll through a file either horizontally or vertically. There are several basic ways to scroll through a file: use the scroll bar, use cursor movement keys, or use the wheel of an intelligent mouse. Each document window has both a horizontal and vertical scroll bar.



Vertical and horizontal scroll bars

Using the Scroll Bars

The vertical scroll bar is located on the right side of a document window. A scroll bar consists of scroll arrows at either end of the bar and the scroll box that moves along the bar. Just as in other applications, it allows you to scroll up and down in a file. The distance you move in a file depends on how you use the scroll bar or how far you move the scroll box.

If you click a scroll arrow, you will move one line at a time. If you drag the scroll box, you can move multiple pages. If you click the scroll bar itself between the scroll box and either of the arrows, you move up or down by one page. The status bar always indicates the line number of your current position.

If your file is wider than your document window, you can scroll left or right by using the horizontal scroll bar located at the bottom of the window. It works the same way as the vertical scroll bar, except it moves left and right.

Using the Intelligent Mouse

If your mouse has a wheel as well as the usual buttons, you have an “intelligent” mouse. The wheel provides an easy way to scroll through a file without moving your pointer to one of the scroll bars. Simply rotate the wheel to move the displayed text.

Qedit recognizes this scrolling method. You can adjust the speed of the scroll by setting the **Mousewheel Acceleration** slider in the **Preferences** dialog box. If you bring the slider all the way to the left, Qedit scrolls slowly for each turn of the wheel. If you bring the slider to the right, Qedit scrolls quickly.

Using Cursor Movement Keys

Another way to move through a file is to use cursor movement keys. Your PC keyboard has the following eight cursor movement keys:

<u>To move</u>	<u>Press this key</u>
one line up	↑
one line down	↓
one character left	←
one character right	→
beginning of the line	Home
end of the line	End
one page up less one line	Page Up
one page down less one line	Page Down

Cursor movement keys can be combined with other keys to provide other shortcuts. See “Key Combinations and Functions” on page 200.

Setting Markers in a File

Marking a Position

Shortcut: F11 key

For each document, Qedit can save up to the last 10 markers at a time. These markers can be an insertion point or a block of selected text. Qedit uses a Last In/First Out (LIFO) stack to save these markers. To mark a position, select the **Save position** command in the **Mark** menu.

For example, if you put markers on lines 2, 10 and 8, and then go to line 20, the cursor returns to line 8, 10 and 2, respectively, when you use the **Go to previous position** command consecutively.

If you save more than 10 markers in the file, you will lose the oldest marker. You will also lose a marker if you delete or cut-and-paste the text in which a marker is located.

Returning to a Marked Position

Shortcut: F12 key

Although you can save up to 10 markers in a file, you can return only to the previously saved marker in the list each time you select the **Go to previous position** command in the **Mark** menu. If you have not saved a marker, this command is unavailable. If you want to mark the position to which the caret returned, you have to use the **Save position** command again.

Exchanging Current Position with Last Marker

Shortcut: CTRL+F11

To return to the last marker without losing your current position, you can use the **Exchange position** command in the **Mark** menu. This switches your insertion point with the last saved marker, as if you had used **Save position** and **Go to previous position** simultaneously.

Using the Last Marker to Select Text

Shortcut: Shift+F12

Use the **Select to Saved Position** command to quickly select text between the insertion point and the position of the last marker. Find the start of the block and save the position with the **Save Position** (F11) command. Find the end of the block anyway you like (scroll, jump, search) and use **Select to Saved Position**. Qedit automatically selects the text between the insertion point and the last marker.

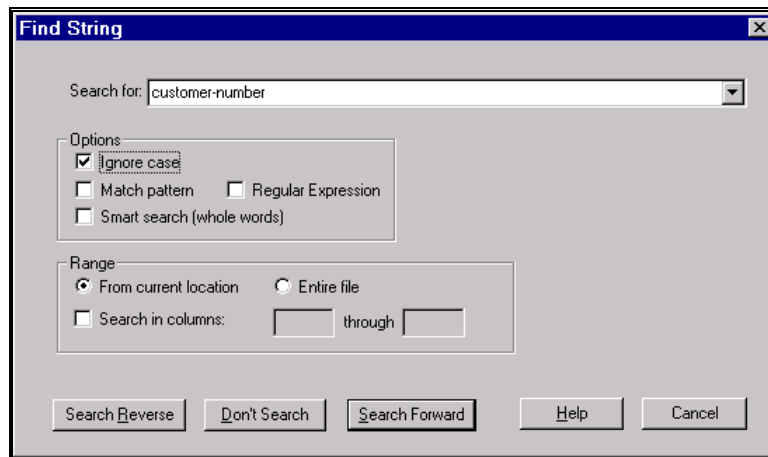
Defining the Terms of a Search

Qedit for Windows provides you with a range of terms to search within a file: pattern matching with substitution characters, case-sensitive matching, searching whole or partial words.

The search functions are as consistent with Windows as possible, but there are some which derive from the experience of Qedit on the host (e.g., pattern matching). If you don't have access to the host-based programs, Qedit for HP e3000 or Qedit for HP-UX, review the corresponding documentation for more details.

You can change the default settings of the search options at any time. This allows you to customize your search and give you the best results. To define the terms of a search,

1. Select **Find** in the **Edit** menu.
2. When the **Find String** dialog box appears, define the terms of your search:



Find String dialog box

- **Search for**

Qedit saves the last 15 search requests you have done. This includes the search string and all options. If you want to recall a search request, scroll through the list using the up or down arrow keys.

If you select some text before calling the dialog box, the selected text is automatically used as a search string unless the previous search used the **Match pattern** or **Regular Expression** options.

- **Ignore case**

Matches the string whether it is in lowercase, uppercase, or mixed case in the file.

- **Match pattern**

Matches patterns of characters instead of specific characters. For example, `@bob@green@` matches any occurrence of the string *bob* anywhere in a line, followed by zero or more intervening characters and then the string *green*. The pattern match control characters are the same as those used in the standard host-based Qedit.

The following characters can be used for other specific pattern searches:

- @ at sign matches zero or more characters of any type
- # number sign matches exactly one numeric character (0-9)
- ? question mark matches exactly one alphabetic character (a-z, A-Z)
- ~ tilde matches zero or more space characters
- & ampersand matches the next character (use `&@` to look for an @ sign)
- ^! circumflex and exclamation point are reserved
use `$^` and `$!` to match them

- **Regular expression**

Regular expressions use powerful syntax allowing precise definitions of your string search. Regular expressions have been around for a long time and are now available on almost any platform. For syntax details, refer to "Appendix C – Regular Expression" on page 243. Note that server version 4.8 or higher is required.

- **Smart search (whole words)**

Search for whole words and ignore embedded words, such as *cell* in *excellent*. Once enabled, this option will not change until you disable it.

The **Smart search** also depends on the Qedit **Language** setting. For example, in COBOL a word can contain a hyphen character, so *word* does not match *word-option*.

- **Range**

You can limit the search to a specific section of the file, specify the starting point of the search, and the columns to search.

3. When you have finished selecting your options, you can proceed in the following three ways:

- Click **Search Forward** to search from the specified starting point towards the end of the file or **Search Reverse** to search from the specified starting point towards the beginning of the file.
- Click **Don't Search** to save the search options, but not perform the search.
- Click **Cancel** to return to the previous search options and not perform the search.

Common Search Operations

Finding Consecutive Occurrences of a String

To search for a text string,

Shortcut: CTRL+F

Toolbar: 



Find Previous is on the left and **Find Next** is on the right.

1. Select the **Find** command in the **Edit** menu.
2. When the **Find String** dialog box appears, define the terms of your search. See “Defining the Terms of a Search” above for a list of terms.
3. Click **Search Reverse** or **Search Forward** to perform the search operation. If Qedit finds a match, it highlights the first occurrence of the string and displays the page around it.

To repeat the same search for the next occurrence of the string, use the **Find Again** command in the **Edit** menu or use the CTRL+G shortcut. To find the next occurrence, use the **Find Next** tool.

To find the previous occurrence, use the **Find Previous** command of the **Edit** menu, use the **Find previous** tool, or type the CTRL+R shortcut.

Finding the First and Last Occurrence of a String

To find the first or last occurrence of a string in a file,

1. Select the **Find** command in the **Edit** menu.
2. To find the first or last occurrence of a string in a file, enable the **Entire file** option instead of the **From current location** option.
3. Define the remaining terms of your search. See “Defining the Terms of a Search” above for a list of terms.
4. Click **Search Reverse** or **Search Forward** to perform the search operation. If **Entire file** is enabled and you select **Select Reverse**, Qedit finds the last occurrence of the string. If you select **Select Forward**, Qedit finds the first occurrence of the string.

Finding a Highlighted Word

To find the next occurrence of a highlighted word, use the **Find "highlighted"** command in the **Edit** menu. To find the previous occurrence of a highlighted word, use the **Find previous "highlighted"** command in the **Edit** menu. These commands use the terms defined in the last **Find** operation.

NOTE: The **Find previous** and **Find next** buttons always look for the last string defined in a search operation, not the currently highlighted text.

Limiting a Search to Part of a File

To limit a search operation to part of a file,

1. Highlight the section you want to search.
2. Select the **Find** command in the **Edit** menu.
3. When the **Find String** dialog box appears, you will see a new option in the **Range** box: **Selection Only** replaces **From current location**. Enable this new option.

Shortcut: CTRL+F

Toolbar: 

4. Define the remaining terms of your search. See “Defining the Terms of a Search” above for a list of terms.
5. Click **Search Reverse** or **Search Forward** to perform the search operation. **Search Forward** finds the first occurrence of the string in the selection. **Search Reverse** finds the last occurrence in the selection.

Searching in Columns

To search for a string in certain columns of a file, enable the **Search in columns** option and enter the column numbers in the appropriate text boxes. Column numbers are relative to the **Language** of the file. For most languages, the first column is one. For others, such as COBOL, the first column is seven.

The left column number is required but the right column is optional. If the right column number is not specified, Qedit assumes the search starts at the left column and continues to the end of the line. The column numbers specified are inclusive. That is, if you enter five as the left column and seven as the right column, Qedit searches three columns: five, six and seven.

Finding a Line Number in a Host File

Lines in a file are identified by numbers. All files, local and host, have relative numbers also known as record numbers. That is, lines are numbered from one and incremented by one.

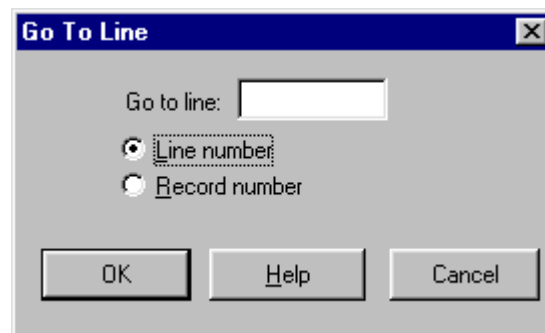
In addition to record numbers, host files often have absolute line numbers. That is, each line has explicit sequence numbers that are stored in ASCII. Absolute line numbers do not necessarily match the line position relative to the file. For example, line number 100 might actually be the 150th line of the file. Absolute line numbers can also have up to three decimals (e.g., 15.25, 217.147).

Although Qedit does not display line numbers in host files, it does allow you to find one. This is very useful when you are working on a server report, such as compiler errors output, which refers to lines by their absolute line number.

To find an absolute line number, select the **Goto line** command in the **Mark** menu.

Shortcut: CTRL+J

The Line number and Record number options are available only for files that have both.



Go To Line dialog box with absolute line number option

When the **Go To Line** dialog box appears, enter the line number and click **OK**. Qedit automatically selects the **Line number** option if the file is numbered. If the file is unnumbered, Record Number is selected.

Finding a Record Number

Shortcut: CTRL+J

Lines in a file can also be referenced by a number relative to the beginning of the file. These ordinal numbers, which are called record numbers, are available in all Qedit files. To find a record number in a file, use the **Goto line** command in the **Mark** menu, enter the desired number and, if given a choice, enable the **Record number** option. For more information, see “Record Numbers” on page 127.

You can also move around in host files using absolute line numbers.

Common Replace Operations

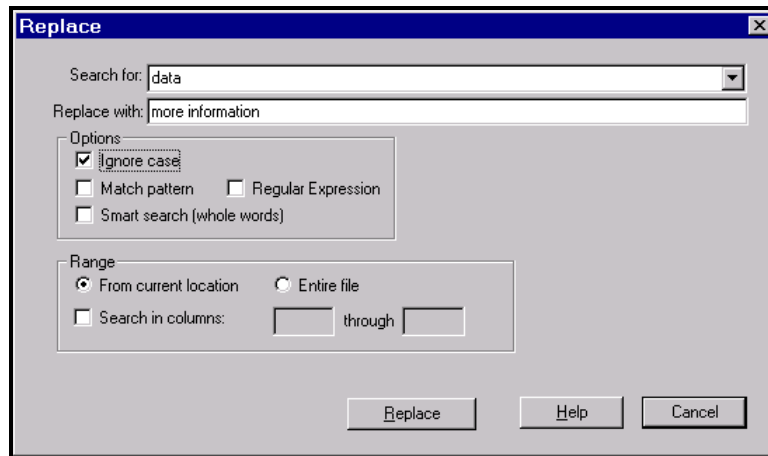
Qedit’s search functions are further enhanced with a replace function. Combined, search and replace can provide you with powerful editing capabilities necessary for global changes to a file.

Replacing Occurrences of a String

To find occurrences of a string from a certain point and replace them with another string,

Shortcut: ALT+E,R

1. Select the **Replace** command in the **Edit** menu.
2. When the **Replace** dialog box appears, enter the string you want to find in the **Search for** text box and the new string in the **Replace with** text box.



Replace dialog box

3. Define the terms of your search operation. You get the same search options as on a **Find** command.

If you have not limited your search to selected text, the default start position of the search is the cursor’s current position. Qedit finds all occurrences of the string from that point to the end of the file and replaces them.

If you have limited your search to selected text, Qedit replaces the string only in the selection.

4. Click **Replace** to perform the operation.

Replacing All Occurrences of a String

To replace all occurrences of a string in a file, select the **Replace** command in the **Edit** menu. When the **Replace** dialog box appears, enable the **Entire File** option in the **Range** box. In this case, Qedit finds all occurrences of the **Search for** string defined by the terms of your search, and replace them with the new string in the **Replace with** text box.

Replacing a String Based on Pattern Matching

You can use the **Match Pattern** option to search for generic strings. Although this can be powerful, it can also be dangerous. Let's say there is a file with the following sentence:

```
Start walking and you will get home in the end.
```

You can enter “@Start@end@” in the **Search for** text box and “Wrong Way!” in the **Replace with** text box, enable **Match Pattern** and select **Replace**. Qedit will search for a line that has the words “Start” and “End” anywhere, with any number of characters before, in between or after. Qedit then puts the replacement string, starting on the position of the *S* in “Start” all the way to the *d* in “End.” In this example, the whole sentence gets replaced with “Wrong Way!”

Replacing a String Using Regular Expressions

Just as a pattern match can be used in a **Replace** operation, you can also use regular expressions. Qedit replaces the whole string matching the regular expression. Regular expressions offer additional functionality in the replacement string. The feature is known as backreference substitution. This allows extraction of regular expression subpatterns from the original text and re-insert them in the replacement string.

Customizing Your Search Options

The caret, cursor and the insertion point are all the same thing.

Allowing the Caret in Undefined Areas

Normally, Qedit does not allow the cursor anywhere to the right of the last character on a line. If you try to enter a character in this area, Qedit automatically moves the cursor to the right-most character on the line. This is Qedit's way of preventing an accidental extension of a line in files where spaces at the end of a line are significant.

You can change Qedit's default so that it automatically extends spaces between the last character on the line and a newly inserted character or a RETURN key beyond the last character. Simply select **Preferences** from the **Options** menu. When the **Qedit Preferences** dialog box appears, enable **Caret allowed in undefined areas**. When this option is enabled, you will be able to put the insertion point beyond the last character of a line.

If you browse through a file while this option is enabled, the cursor remains in the same column no matter where the text ends in a line. Qedit does not adjust left or right. If the option is disabled, however, Qedit keeps the cursor in the same column only if the text extends that far. For shorter lines, the cursor is moved to the end of text, adjusting left or right as needed to always keep the text and cursor visible.

If you start typing while the cursor is in the undefined area, Qedit automatically fills the area up to that point with blanks. If you also have enabled **Preserve trailing blanks**, you might start getting error messages as you insert text.

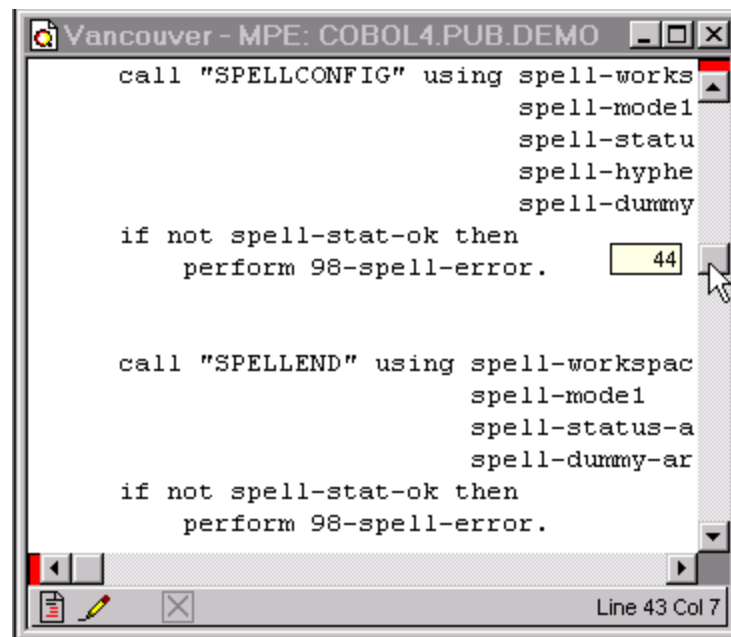
Automatically Marking Successful Searches

To make it easier to move around in a file, you can set markers using the **Save position** command of the **Mark** menu. When the **Automatically save position on search** option on the **Searching** page of the **Preferences** dialog box is enabled, Qedit automatically marks a position after a successful **Find** operation, if the search string is different than the previous one.

Scrolling Method for Host Files

Depending on your network or PC, scrolling through host files using the scroll box can be slow. This is because Qedit is requesting new lines from the server as you move the scroll box. You might not actually need these lines, however, so Qedit now provides the **Use live scrolling for server files** option of the **Preferences** menu.

By default, this live scrolling option is disabled. This means Qedit will not request new lines from the server until you release the scroll box. While you move the scroll box, Qedit displays the record number corresponding approximately to that location in the file. The record number appears in a box to the left of the scroll bar. Also, the text already displayed in the window does not change or move until you release the scroll box.



The record number appears in a small box during passive scrolling.

When enabled, Qedit requests lines as you move the scroll box. If you have a fast network or if most of the file is already in the cache, you will not see any delays. The text in the document window will move up and down as you move the scroll box.

Setting Default Search Options

You can save your favorite search options (**Search entire file**, **Ignore case**, **Smart search**) as the default settings for a **Find** operation. To set your defaults, select **Preferences** in the **Options** menu. When the **Preferences** dialog box appears, enable your choice of options in **Search Defaults** box.

Search options have three different settings. If an option is disabled (e.g., **Smart search**), the corresponding option in the **Find String** dialog box will also be disabled. If an option is enabled (e.g., **Ignore case**), the corresponding option in the dialog box will also be enabled. If an option appears dimmed (e.g., **Search entire file**), the corresponding option in the dialog box will use the setting in the previous search.

Copying & Printing Files

Overview

Copying and printing files is easy with Qedit for Windows. You can copy files across platforms with a simple cut-and-paste or use the standard Windows **Save As** command. Copy a local file to another local file, a local file to a host file, or a host file to the same or another host file or to a local file. Print a host file on any printer configured on the host connection or on a local printer. You can choose from a variety of print options to customize your printout.

Copying a File

By using either cut-and-paste or the **Save As** command, Qedit for Windows can copy files across all the platforms to which it is connected.

Copying Local Files

Copying a Local File to Another Local File

You can use the standard Windows **Save As** command to copy a local file to another local file.

1. Open the local file you want to copy.
2. Select **Save As** from the **File** menu.
3. When the **Save As** dialog box appears, enter the new file name. If you want to put the new file in a different directory, you can select another directory in the file structure.
4. Click **Save**.

Copying a Local File to a Host File

You can make a copy of a local file and put it on a host by cutting and pasting. To copy a local file to a host,

1. Select the **File** menu, point to **Open**, and click **Local**.
2. Select **Select all** from the **Edit** menu to copy its contents.
3. Select the **File** menu, point to **New**, and click **Host** to create a new host file.

4. When the **New Host** dialog box appears, define the characteristics of the new host file and click **New**.
5. When the new document window appears, use the **Paste** command in the **Edit** menu to paste the local file into the new host file.

NOTE: The Clipboard has a limited capacity. See “Workarounds to Clipboard Cache” on page 95.

Copying Host Files

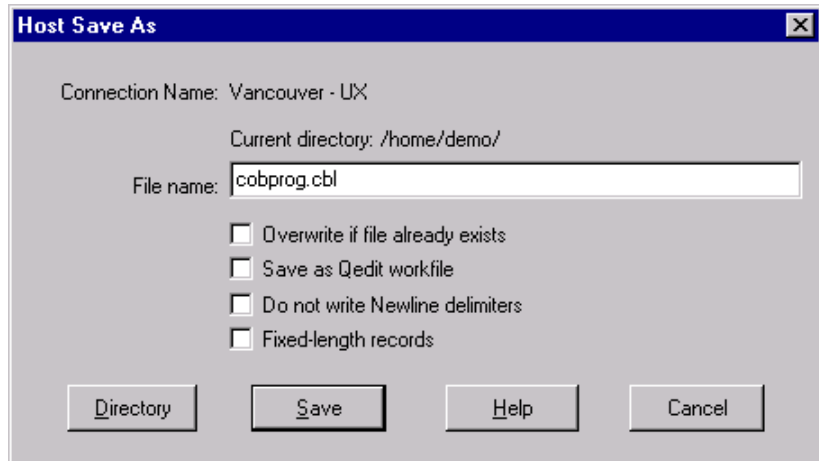
Copying a Host File to the Same Host

There are times when you want to modify a file, but you don’t want to lose the last version of the file. To copy a host file to the same host is a two-part process: first open a copy of the file, then save it with a new file name. You can also use this process to rename a host file.

To copy a host file to the same host,

1. Select the **File** menu, point to **Open**, and click **Host**.
2. When the **Open Host File** dialog box appears, select the file that you want to copy.
3. Enable the **Open a copy** option to open a copy of the file you selected. The **Open a copy** option is available in both the **Open Host File** and **Directory** dialog boxes. If you select multiple files in the **Directory** dialog box, the **Open a copy** option will apply to all of them.
4. Click **Open** to open a copy of the file. Qedit will open the copy and you can edit it without modifying the last version of the file.
5. Select the **Save As** command in the **File** menu.
6. When the **Host Save As** dialog box appears, give the file a new name. Qedit will save this file in the current directory. If you want to save the file in a different directory, enter a path to another directory including the new file name. See “Notation in Host Operating Systems” on page 67 for more details.

If you do not remember the path name to the new location, you can use the **Directory** button and go to the **Directory** dialog. There, you can navigate the directory structure until you get to the desired location.



Host Save As dialog box

- If the new file name already exists, Qedit asks if you want to overwrite the existing file with the new one. Check whether you are really in the directory you think you are before clicking **OK**. The ESCAPE key or **Cancel** button is always safer!

If you already have a file with the new name, Qedit gives you an option to overwrite its contents, eliminating the confirmation message.

- Qedit also gives you an option to save the new file as a Qedit workfile.
- Enable **Do not write Newline delimiters** to save the file without any Newline delimiter. This option is automatically enabled if the **Length** option has been specified on the **Open Host File** dialog box. This option is only available with the HP-UX server version 5.3.13 or later.
- Enable **Fixed-length records** if you wish to save the file with lines of equal length. This option is automatically enabled and can't be disabled if the `CobolFixed` server setting is enabled and you are editing a Cobol source file. This option is only available with the HP-UX server version 5.3.13 or later.

7. Click **Save** to finish saving the copied file.

If you have trained your fingers to use shortcut keys (e.g., CTRL+S), you won't have to worry about wiping out the existing version of a file because you work with a copy of it.

You can also use this process to rename a host file. After saving the copy, use the **Host Command** dialog box and purge the old version. Of course, you could use the **Host Command** dialog box and enter a `REname` (MPE) or `mv` (UNIX) command to get the same result.

You can also make a copy of a host file by using the **Select All**, **Copy** and **Paste** commands of the **Edit** menu. The **Save as** method is more productive because you only need a single command that uses less resources on the client and the server.

Copying a Host File to Another Host

You can also copy a host file from one host to another.

1. Open the host file you want to copy.

2. Copy its contents with the **Select All** command in the **Edit** menu.
3. Create a new file on another host.
4. Paste the contents of the original host file into the new host file.

Copying a Host File to a Local File

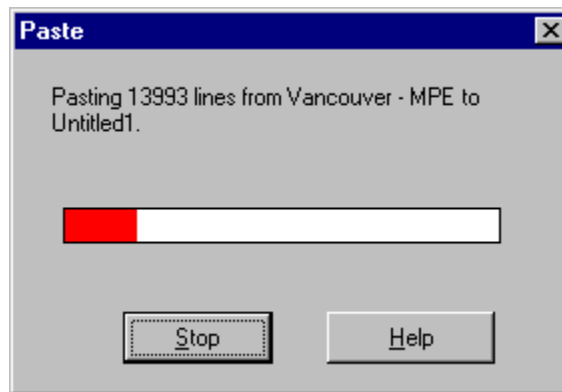
There is no single command that allows you to create a local copy of a host file. You have to use the **Copy** and **Paste** commands.

1. Select the **File** menu, point to **Open**, and click **Host** to open the host file.
2. Select the text you want to copy. If you want to copy everything, you can use the **Select all** command in the **Edit** menu.
3. Copy the text to the Clipboard by using the **Copy** command in the **Edit** menu.
4. Create or open the local file into which you want to copy the text.
5. Paste the contents of the Clipboard by using the **Paste** command in the **Edit** menu. As Qedit pastes, it displays the **Paste** progress dialog box.

Shortcut: CTRL+A

Shortcut: CTRL+C

Shortcut: CTRL+V



Paste progress dialog box

Converting a File to Qedit Format

A Qedit workfile is a host file with a special format that is valid only in a Qedit work environment. When you edit a Qedit workfile, all changes are posted to the host disc drives as soon as you enter them.

NOTE: Ordinary programs cannot read Qedit workfiles directly without modifications. While an MPE host has tools available that allow easy access to Qedit workfiles, an HP-UX host does not. For this reason, you might want to save Qedit workfiles as text files when you are finished editing them.

To convert a text file into a Qedit workfile,

1. Open the file you want to convert to the Qedit format.
2. Choose the **Save As** command in the **File** menu.
3. When the **Host Save As** dialog box appears, choose the **Save as Qedit Workfile** option.
4. Enter the new file name. Qedit will save this file in the current directory. If you want to save the file in a different directory, enter a path to another directory

including the new file name. See “Notation in Host Operating Systems” on page 67 for more details.

5. Click **Save**.

Convert Qedit Workfile to Text Format

To convert a Qedit workfile into a text file,

1. Select the **File** menu, point to **Open**, and click **Host**.
2. When the **Open Host File** dialog box appears, select the file that you want to copy.
3. Enable the **Open a copy** option to open a copy of the file you selected. The **Open a copy** option is available in both the **Open Host File** and **Directory** dialog boxes. If you select multiple files in the **Directory** dialog box, the **Open a copy** option will apply to all of them.
4. Click **Open** to open a copy of the file.
5. Select the **Save As** command in the **File** menu.
6. When the **Host Save As** dialog box appears, give the file a new name. Qedit will save this file in the current directory. If you want to save the file in a different directory, enter a path to another directory including the new file name. See “Notation in Host Operating Systems” on page 67 for more details.
7. Click **Save**.

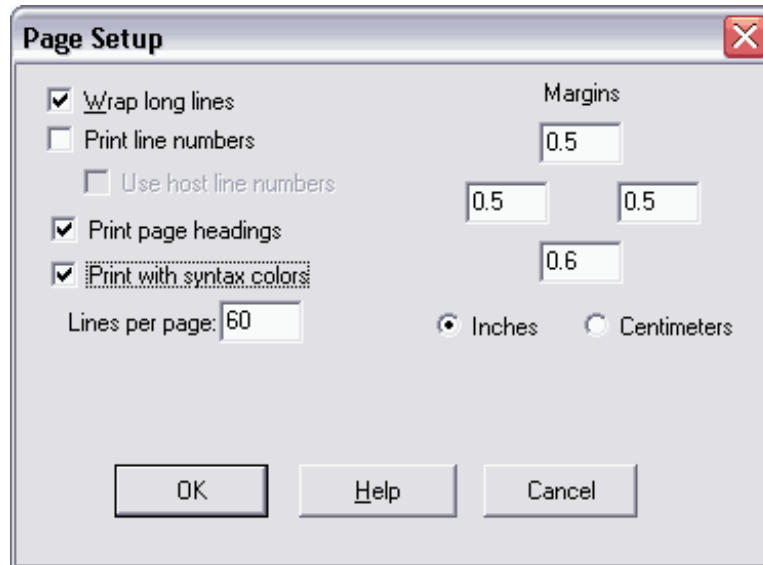
Printing on a Local Printer

Qedit can access local and remote equipment, so you can print a hard copy of a file on the printer attached to your PC or you can send the file to a printer configured on the host (host files only).

Print Options on a Local Printer

The Page Setup command is also accessible by clicking and holding the Print button.

You can send a local or host file to the printer attached to your PC or normally available from Windows. You can specify the print options by using the **Page Setup** command in the **File** menu.



Page Setup dialog box

When the **Page Setup** dialog box appears, you can specify the print options for the local printer. These settings do not control the print options for the host printer.

- **Wrap long lines**

This option is enabled by default. If lines are too wide to fit on the page, Qedit fits as much as possible on each line and then wraps the remaining text on the next line. For most languages Qedit wraps at word boundaries. Although Qedit does wrap host files with a **Language** of DATA, it does not necessarily wrap them at word boundaries. All local files are assigned a hidden **Language** of SPL.

Continuation lines are indented four (4) characters from the left margin so that they are easier to identify.

If you disable this option, Qedit fills each line and truncates the text at the next character, even if it is in the middle of a word.

- **Print line numbers**

You can request to include line numbers on the printed page. Note that these are really records numbers, a sequential number that represents a line relative to the beginning of the file.

- **Use host line numbers**

If you asked to get line numbers and you are dealing with a host file, you can ask for the actual line numbers and not the sequential record numbers.

- **Print page headings**

By default, Qedit prints a one-line header with the PC's current date and time, the file name, and the page number. If all this information does not fit on a single line, Qedit replaces some of the characters from the middle part of the file name with an ellipsis.

If you do not want to include the page header, simply disable this option.

- **Print with syntax colors**

This function is not available at the moment.

By default, Qedit prints in black only. To print using the current syntax colors, enable this option. Qedit uses the same colors as those used on the screen. Keep in mind that certain screen colors might not show well on paper (white background) if the selected screen background is dark. For this reason, Qedit changes anything in white to black.

- **Lines per page**

This setting specifies the number of printed lines per page. By adjusting the number of lines per page, you can change the font size. The more lines per page you request, the smaller the font and the more characters per line. This setting can be used to remove wide lines that wrap on a page. Another way to fit more information on a page is to reduce the settings for the left and right margins.

Qedit uses your document font as your print font (or the print driver will use the closest approximation). If you want to print the file in a different font, you first have to change the font in your document. Select **Properties** from the **File** menu and click **Set font**.

- **Margins**

The four text boxes under **Margins** indicate the amount of space in the margins of the page. Each text box represents a side: top, bottom, left and right.

In the example above, the top, left and right margins are set to 0.5 inch. The bottom margin is set to 0.6 inch.

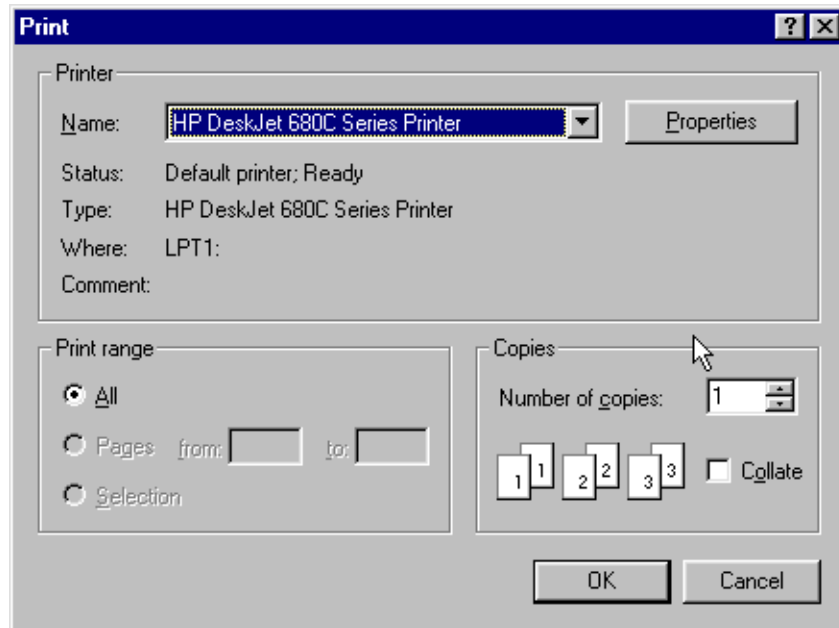
- **Inches and Centimeters**

These buttons indicate the unit of measure used with the margin settings.

Printing on a Local Printer

To print on a local printer,

1. Select the **Print** command in the **File** menu.
2. When the **Print** dialog box appears, specify the pages you want to print. By default, Qedit presumes you want to print the whole file. If you have selected some text, the **Selection** option is enabled automatically.



Windows Print dialog box

3. If you want to send the file to another printer, select it from the **Name** list.
4. Select any other options you want to use (**Number of copies** or **Properties**).
5. Click **OK** to print the file.

To send the output to a printer on a host, see "Printing on a Host Printer" on page 154.

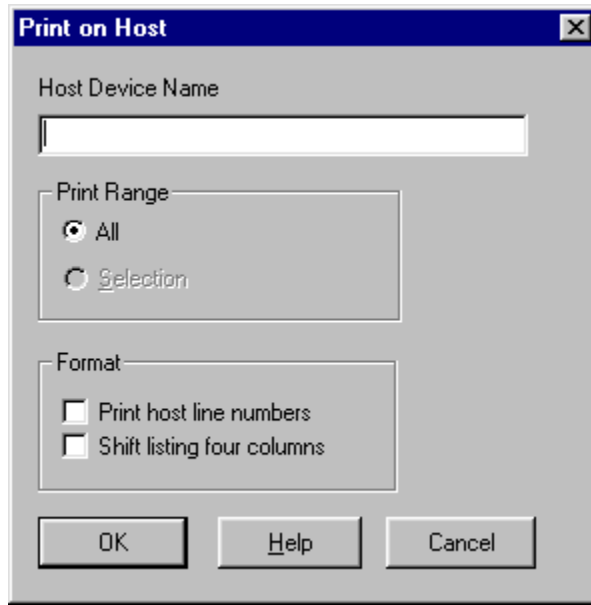
Printing on a Host Printer

If you are working on a host file, you can print its content to a local printer or one connected on the host computer. You cannot print a local file directly to a host printer.

Print Options on a Host Printer

Options in the Print on Host dialog box are only available with version 4.6.54 (or later) of the server.

You can also ask Qedit to print a host file on a printer connected to the host. You can specify the print options by using the **Print on host** command in the **File** menu. The **Print on host** command is not available if you are editing a local file.



Print on Host dialog box

When the **Print on Host** dialog box appears, you can specify the print options for the host printer.

- **Host Device Name**

Choose the destination host printer in this text box. The syntax for the device name is specific to the host you are connected to. Look at “Changing the Default Printer in MPE” below for other ways to print to a host printer.

- **All or Selection**

By default, Qedit prints the complete file. However, if you have highlighted some text, Qedit will print only the selected lines. You can override this by enabling the **All** option.

- **Print host line numbers**

You can print the host line numbers along with the text by selecting this option.

- **Shift listing four columns**

You can change the amount of room in the left margin by using this option to shift four columns to the right.

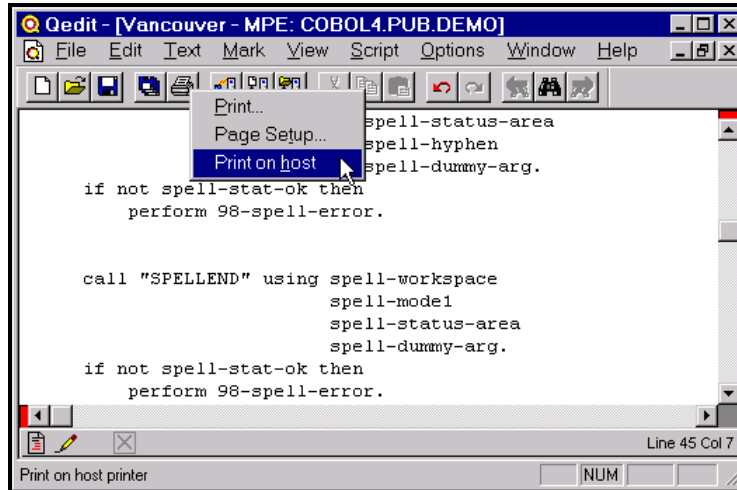
Changing the Default Printer in MPE

Qedit allows you to send output to a printer connected to the host computer. On MPE, the default device class is LP. If you want to send your output to a different device, you need to enter a file equation in the server configuration file.

```
if Qedservmode <> 0 then
  :file lp;dev=serverlp
endif
```

Printing on a Host Printer

Instead of using the **Print** command in the **File** menu, you can click and hold the **Print** button to display a shortcut menu from which you can select print-related commands.



Click and hold the Print button to get the print shortcut menu.

Saving & Closing Files

Overview

You can quickly save and close files in Qedit, both explicitly and implicitly. The familiar Windows interface makes saving local files an easy routine to develop. Changes to Qedit workfiles are immediately posted to the hard disc, so your work is automatically saved. Closing files is equally simple. Whether you close individual files or the program itself, you are prompted with saving options before a file is closed.

Saving a File

Which files do you need to save?

Just because you typed some lines into a new or existing file, does not mean that your work has been saved. If there is a problem with the PC or the server, or the TCP/IP connection between them stops, it is possible that you will lose the work you have done.

That is why Qedit for Windows has a save function accessible in many different ways. Most local and host files have to be saved explicitly, individually or all at once. Local files also have an auto-save feature. Of course, if you are using Qedit workfiles, you do not have to worry about explicitly saving all your changes.

Auto-Saving Local Files

If you want to worry about saving your work, you can automatically save changes to local Qedit files at regular intervals. See “Saving Changes at Regular Intervals” on page 162.

Saving a Qedit Workfile

If you are working on a saved Qedit workfile, you do not need to perform an explicit save to protect your work, because workfiles are directly connected to the host so that changes are posted immediately. If you are editing a copy of a Qedit workfile, you must save the copy by performing a **Save As** command. If the connection is lost before you save the copy, only the changes that were made after the file was

copied might be lost. Under most circumstances, it is possible to recover your changes, see "Recovering a File" on page 219.

If you are working on a Qedit workfile and you want to save it as a regular file, you need to open a copy of the workfile and then save it under a new name. If you do not open a copy of a workfile, a **Save As** command simply renames the workfile.

Saving an Existing File (Local and Host)

Shortcut: CTRL+S

To save changes to an existing file, whether host or local, use the **Save** command. Save is also one of the standard buttons on the toolbar. This menu choice is unavailable if you have not made any changes since the last save, if your file does not have a name yet (**Copy** or **New**), or if you opened the file with **Read Only** access.

If the file has been modified by someone else since you opened it or last saved it, Qedit will warn you and suggest not doing the **Save**. This option can be disabled in the **Preferences** dialog box in the **Options** menu.

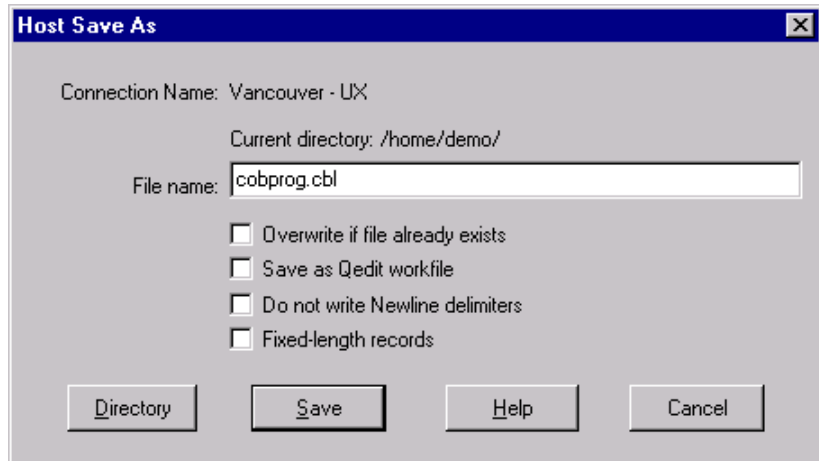
The **Load local files ... K into memory** option also affects how Qedit saves changes to a local file. If you enable this option, modifications to a local file that is smaller than the configured threshold are saved only when you perform a save operation. When Qedit opens these files, it reads the text into your PC's memory and closes the original file. If, however, a file is larger than the configured threshold or this option is disabled, Qedit treats the file like a host file and continually saves your changes to the original file. For more information, see "Loading Local Files into Memory" on page 86.

Saving a New Host File

To save a new host file once you have created it,

1. Select the **Save** or **Save As** command of the **File** menu.
2. When the **Host Save As** dialog box appears, enter the new file name. Qedit will save this file in the current directory. If you want to save the file in a different directory, enter a path to another directory including the new file name. See "Notation in Host Operating Systems" on page 67 for more details.

If you do not remember the path name to the new location, you can use the **Directory** button and go to the **Directory** dialog. There, you can navigate the directory structure until you get to the desired location.



Host Save As dialog box

- If the new file name already exists, Qedit asks if you want to overwrite the existing file with the new one. Check whether you are really in the directory you think you are before clicking **OK**. The ESCAPE key or **Cancel** button is always safer!

If you already have a file with the new name, Qedit gives you an option to overwrite its contents, eliminating the confirmation message.

- Qedit also gives you an option to save the new file as a Qedit workfile.
- Enable **Do not write Newline delimiters** to save the file without any Newline delimiter. This option is automatically enabled if the **Length** option has been specified on the **Open Host File** dialog box. This option is only available with the HP-UX server version 5.3.13 or later.
- Enable **Fixed-length records** if you wish to save the file with lines of equal length. This option is automatically enabled and can't be disabled if the `CobolFixed` server setting is enabled and you are editing a Cobol source file. This option is only available with the HP-UX server version 5.3.13 or later.

3. Click **Save** to finish saving the file.

Saving a New Local File

To save a new local file once you have created it,

1. Select the **Save** or **Save As** command in the **File** menu.
2. When the **Save As** dialog box appears, give the file a name and select the directory in which you want to save it.
3. Click **Save** to finish saving the file.

If a file has not been saved yet or you use the **Save As** command on an unsaved file, a **Save As** dialog box will appear so that you can select the drive, folder and file name.

Saving All Opened Files

If you want to quickly save your changes to all the files currently opened, just select the **Save All** command in the **File** menu. If you have newly created files, Qedit will prompt you for new file names. If you were working with existing files, the changes are simply written to disk.

Closing a File

Explicit vs. Implicit Close

When you close a file, Qedit posts all changes to it, releases any exclusive-access claims you have on it, and closes its document window. Of course, there is an explicit **Close** command on the **File** menu, but there are also implicit closes. Implicit closes occur when you **Exit** from Qedit or when you **Disconnect** from a host.

Closing a File

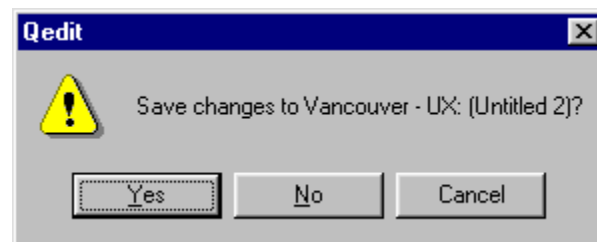
In both local and host files, you can explicitly close the currently active edit window by using the **Close** command in the **File** menu. There is no corresponding toolbar button. In Windows 95, you can use the **Close** button in the top right corner of the window. You can also use the CTRL+F4 shortcut. The edit window you closed should disappear and, if you have other files opened, another one will take its place on your screen.



Close button in Windows 95 has an X.

What if there are unsaved changes?

If you have made changes to the file and not saved them, Qedit will ask you if you want to save your changes.



Save confirmation dialog box

You have three ways to respond:

- **Yes** - Qedit saves the file. If the file has not been named yet, Qedit also prompts you for a file name as if you had selected the **Save As** command.
- **No** - Qedit closes the file and your latest changes are lost.
- **Cancel** - Qedit cancels the close operation and brings you back to the edit window.

What happens to connections when you close a file?

When you close a host file, Qedit usually leaves your connection to that host active, in case you want to open another file on the same host. For more information, see “Closing a Connection” on page 59 and “Closing Connections without Open Files” on page 59.

Forcing a Close on a Host File

Qedit always keeps track of what is going on with the server. The server always acknowledges each transaction that it receives. If a transaction is critical, Qedit will not continue until it receives a confirmation from the server.

When a file is closed, for example, the document window remains on the screen until the server confirms that the file has been closed properly. If there is a problem and Qedit does not receive an acknowledgment, a document window might remain open. If a user tries to repeat closing the file, Qedit will display a **Force Close** dialog box because it is waiting to finish processing the first request. Another symptom of an incomplete close operation is the continued appearance of transmission arrows. If Qedit receives another close operation while an earlier close operation is still in progress, you might see one of the following error messages in the **Force Close** dialog box:

```
Error closing connection name: filename. Close anyway?  
Some information may be lost.
```

```
Unknown error closing filename. Close anyway? Some  
information may be lost. (Server error error number.)
```

```
Connection name has not finished operation the file.  
Do you want to close anyway? Some information may be  
lost.
```

Select **OK** if you want Qedit to close the file without waiting for the server to respond. As mentioned in the message, you will be interrupting a transaction. If data movement is involved, you run the risk of losing data. You should only select **OK** when you are sure the connection with the server has been broken and the transaction cannot be completed.

If you want to give the transaction a bit more time, select **Cancel**.

Customizing Saving and Closing Files

Automatically Posting Qedit Workfiles on the Server

To post the latest changes to all workfiles when you go to another Windows application, select **Preferences** from the **Options** menu, and enable the **Automatically post Qedit workfiles** option. Qedit offers this option in case another application needs to use the file. If you disable this feature you can lose some changes to the file.

Comparing Timestamps Before Overwriting Host Files

By default, Qedit does not save your edited file on the server if someone else has modified the file after you got your copy of it. If you choose the **Compare Timestamps before Overwriting on Server** option, Qedit allows you to write over other people's work without receiving a warning. To change the default, simply select **Preferences** in the **Options** menu, and enable this option.

Saving Changes at Regular Intervals

Qedit can now automatically save the latest changes to a file at regular intervals. To use this feature, you have to enable the **Save automatically** option in the **General** tab of the **Preferences** dialog box. Then specify the number of minutes between **Save** operations. If you are working with text files (as opposed to Qedit workfiles), your work is regularly saved. If you lose your connection, you only lose your work from the last automatic **Save** operation.

When this feature is enabled, *all changes to all your currently opened files are saved at the same time*. Qedit does not allow you to set different intervals for specific documents.

Interacting with Hosts

Overview

Qedit has the capability to interact directly with MPE and HP-UX hosts. You can execute commands directly on the host and view the results in a dialog box.

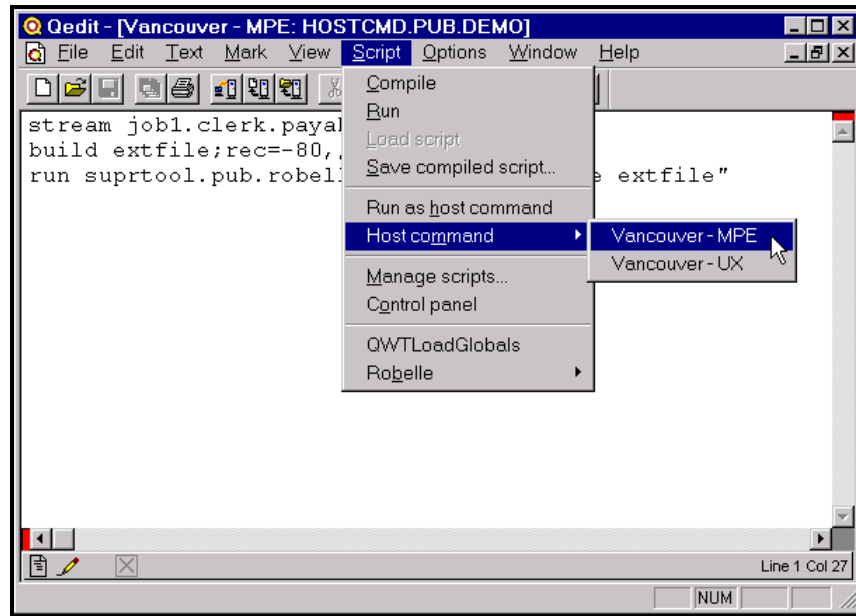
Executing Host Commands

Qedit offers different ways to execute host commands. You can execute them one at a time and see the results immediately. You can create a document with all the commands that you need and have Qedit execute them as one. If you frequently execute the same commands, you can write a script that can be used repeatedly.

The command appears dimmed if the server does not support the feature.

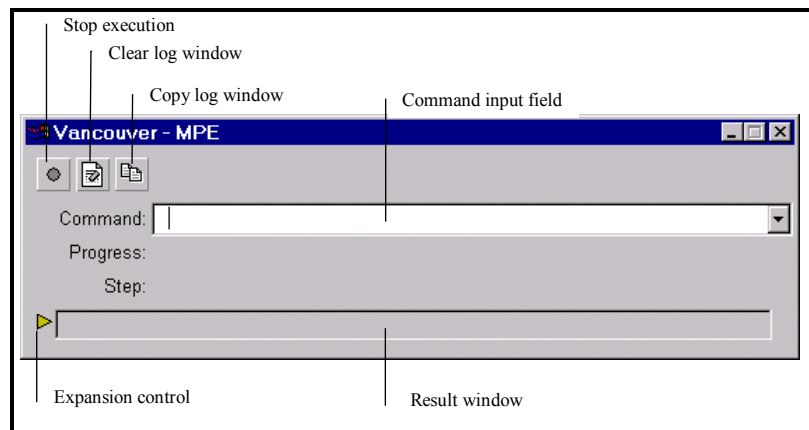
Single-Line Execution

Use **Host command** on the **Script** menu to select the connection on which to execute the host command. Qedit displays only connections that are already open. If you want to execute a command on a different connection, you have to open the connection ahead of time by opening a host file, for example.



Host Command connection list

Selecting a connection from the **Host command** list displays the host command dialog box.



Host command dialog box with single-line result window

It is *very important* to note that the dialog box is not meant to be a terminal emulator. It is simply a place to display information. And even then, output destined to format terminal screens, such as escape sequences, are inoperative here. Nonprinting characters are displayed with small empty squares, as in `_&dJ`. In this example, the square is a placeholder for the escape character.

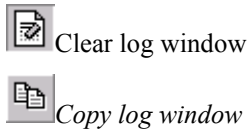
The title bar shows the name of the connection on which commands are executed. You can open multiple host command dialog boxes simultaneously.

Enter the command you want to execute in the **Command** field. You can enter only one command at a time. It can be any command that does not require user input and is normally available on the host via the command interpreter (CI) on MPE or the shell on UNIX. You will not be able to do anything else in Qedit for Windows until

the command has terminated. The Command list box keeps track of the last five commands.

If you want to interrupt a command, click the **Stop** control button. The button is red only when a host command is currently being executed. If there is no command, the button is grey. You might want to interrupt a command if it is taking too long to execute, if it transmits too much information back or if the command is interactive (i.e., prompts the user for information). In the latter case, Qedit's behavior is unpredictable. Typically, Qedit simply waits for more command output. If the **Stop** button does not seem to work, the only other option is to close the connection using the **Disconnect** command of the **File** menu.

Although an interrupt signal is sent to the host, new output lines might still be added to the result window. That's because the server has to interpret the signal and then try to stop the executing command. Although this is fairly fast, the host command still has enough time to generate more output. In addition to this, output lines might already be in transmission buffers. Qedit has to empty these buffers, even though the host command has already terminated. If you interrupt a command, you should not rely on the output because it might be incomplete.



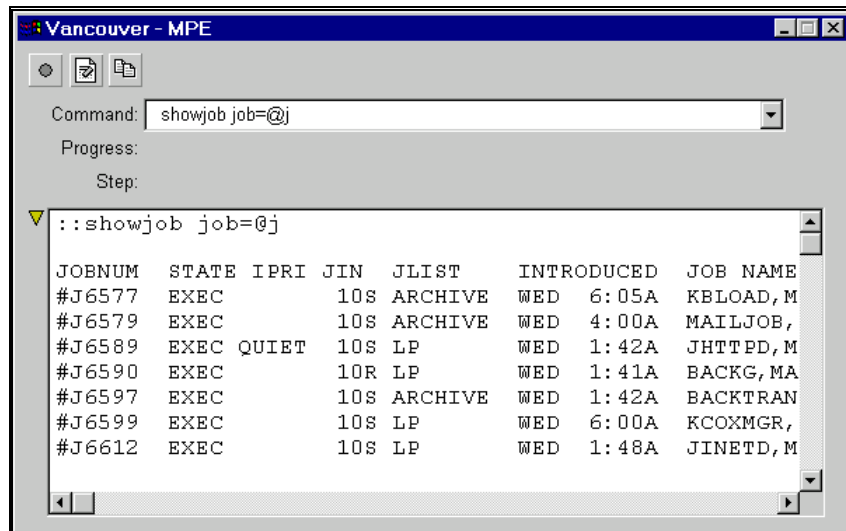
The output log window is not cleared with each command. Instead, output from successive commands is simply appended at the end. If you wish to get rid of previous output text, click on the **Clear Log Window** button. Similarly, if you wish to capture the contents of the output window, click on the **Copy** button. Everything in the log window is then copied to the local clipboard.

The **Progress** field displays information on the execution process. This is an indication on the amount of CPU resources used on the host.

The **Step** field displays information on the progress in a multi-step execution such as a command file or script. It shows the command that is currently being executed.

The host command output is displayed in the result window. By default, the result window only shows one line at a time. That's the last line received from the host.

If there is more than one output line, you can expand the result window by clicking on the expansion control, which is the red arrowhead to the left of the result window.



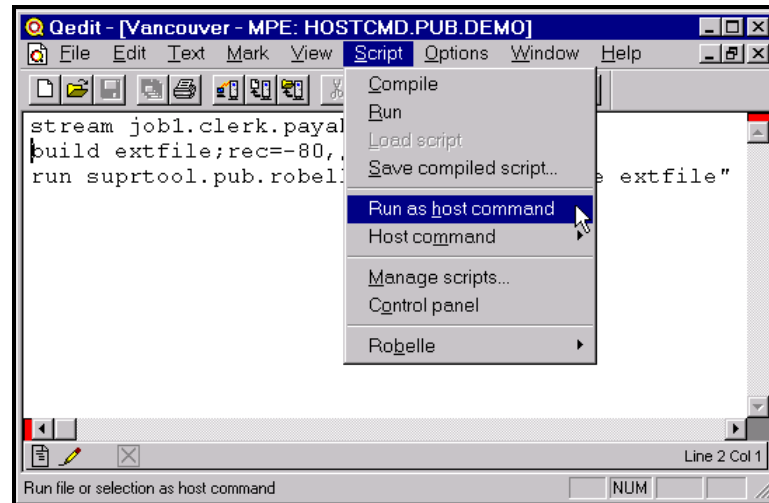
Host command dialog box with expanded result window

The result window turns into a workspace with scroll bars. The expansion control changes to a downward-pointing arrow. You can change the window size at any time. You don't have to do it before executing a command. To return to a single-line window, click the expansion control again.

The result window can hold approximately the last 500 lines.

Multi-Line Execution

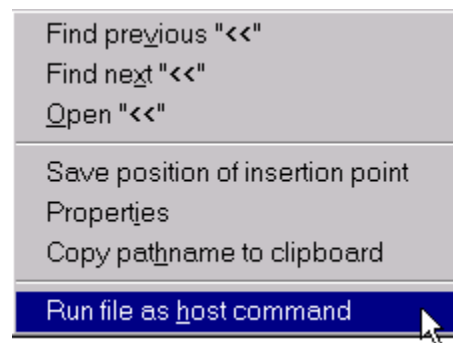
If you are working on a file that contains host commands, you can execute them using the **Run as host command** command of the **Script** menu.



Run as host command from the Script menu

If there is no active selection, the whole file is used as a series of host commands. If there is an active selection, only the selected text is used. The host command output is automatically sent to the corresponding host command dialog box. If the dialog box is closed at the time, Qedit opens it automatically.

For faster access to this feature, you can use the shortcut menu by clicking the right mouse button. The last option of the shortcut menu provides the same functionality.



Run File as host command on the shortcut menu

The file does not have to contain only CI or shell commands. You should think of the file as an MPE job stream or a UNIX shell script. It can contain a mix of operating system commands and program input. For example, on MPE it is perfectly valid to execute the following:

```
run suprtool.pub.robelle
Base store

Get d-sales
List standard
Exit
```

Similarly, on UNIX you can use:

```
sqlplus
scott/tiger
select * from emp;
exit;
```

Disabling Host Command Execution

Although host commands are useful, there are situations where you might want to disable this feature for certain users. This can be done by inserting the following statement in one of the server configuration files:

```
Set Limits Sys Off
```

MPE Commands Execution

Host commands are not executed by the server. They are executed by a separate command interpreter (CI) process.

To execute commands on MPE hosts, the server runs an instance of the command interpreter, `CI.PUB.SYS`, as a child process. The CI process is created with `Stdin`, `Stdlist` and `Stderr` redirected. The commands entered in the host command dialog box or in the host document are passed in as `Stdin`. Any output normally written out by each command to `Stdlist` is captured in temporary file. Similarly, if a command writes information to `Stderr`, these messages are also going to be captured in a temporary file. Both `Stdlist` and `Stderr` are then transmitted to the client where they are displayed in the host dialog box.

Because the CI process is part of the session process tree, it has access to almost everything that makes up the session environment such as MPE variables. However, there are a few things you have to remember.

User Defined Commands (UDC)

User Defined Commands (UDC) are not cataloged thus are not available. Logon UDCs are not executed when the connection is established. Command files and programs are accessible but, since logon UDCs are not executed, the `HPPATH` variable might not be set, MPE might not be able to find the file you are looking for.

If some of the host commands require that the environment be set in a particular way, you have to either

- explicitly set the environment before you start executing the host commands
- change the QEDITMGR server configuration files so the environment is initialized properly. Server-specific settings can be executed conditionally by checking the QEDSERVMODE variable first.

Session Environment

When a connection is established, a new session is started and the server program is run. This is a normal session. It runs under a user within a group and account. It has access to files in the permanent and temporary domains. Temporary files are only

accessible from this connection and nowhere else. Generic files such as \$OLDPASS are also available. You can very well compile and link a COBOL program by entering the following commands in the host command dialog box or in a host file:

```
cob85xl cobol4.src
link $oldpass
run $oldpass
```

Process-dependent Settings

Most changes made to the session environment are available to other processes in the process tree. There are certain commands however that only affect the current process. For example, if you use the `CHDIR` command, the current working directory is changed only for the application it is issued from. As soon as you exit from that application, the CWD returns to its previous value.

This has serious implications for host commands entered in Qedit. If you are using the host command dialog box, each command is executed by itself in its own CI process. If you execute commands straight from a document, a single CI process handles all the commands in the document.

In the `CHDIR` example, the command is executed and forgotten immediately. The next command you enter might fail if it assumes the `CHDIR` is still effective. Worst, it can act upon a different set of files. A `Purge` command might remove something totally different than what you intended.

In this situation, you have 2 options. You can put all your commands, including the `CHDIR` command, in a document and use the **Run as host** command. The other option is to use the **Directory** dialog box and manually change the CWD to the location you want to work from.

Session Initiation and Termination

You will not be able to use a `Hello` command to start a new session. This command implies termination of the previous session which is not allowed. The CI process is started but will not be able to complete the request. The dialog box will appear as if it was waiting on something. To get the control back, simply select the **Stop** button.

You will get similar results if you type `Bye, .`

Duplicate Output Lines

There are cases where lines in the output window of the host command dialog box will appear twice. This would be the case for most applications using a **List** parameter to redirect output data. Most compilers are such applications. A typical compiler command file would be like:

```
COB85XL.PUB.SYS
parm text="", obj="", list="", mast="", new="", info="",
wksp="", xdb=""
```

Compilers usually write some kind of messages to indicate the number of errors, for example. Compile statistics are always written to `Stdlist`. Depending on the compiler directives, you can request additional output. For example, you can ask the compiler to print the source lines as they are compiled. The compile statistics are also printed with these. You can redirect the additional output to another device or



Stop button

file using the **List** parameter. By default, this parameter is directed to `Stdlist`. The compiler is able to determine that the default output and **List** parameter are one and the same. It then assumes there is no need to write the compile statistics twice.

When compiling from Qedit, the CI process in charge of host command execution already has `Stdlist` redirected. The compiler then thinks the default output and **List** parameter are different so it writes the statistics twice.

The only way to remove these duplicate lines is to specify a **List** parameter. You would then get only the compiler statistics. The disadvantage is that you have to manually look at the **List** output. For example, if **List** was pointing to a file, you would have to print the file or open it with Qedit.

UNIX Commands Execution

To execute UNIX host commands, the server starts a new instance of the user's default shell. The commands are then executed inside that instance. Any output is captured and returned to Qedit for display in the host command dialog box.

Since a new shell is created, all UNIX rules and limitations are applicable. For example, you can change an environment variable like `PATH` but, as soon as the process terminates, the changes are discarded.

For example, if you use the `CD` command, the current working directory is changed only for that particular process. As soon as you exit from it, the `CWD` returns to its previous value.

This has serious implications for host commands entered in Qedit. If you are using the host command dialog box, each command is executed by itself in its own shell process. If you execute commands straight from a document, a single shell process handles all the commands in the document.

In the `CD` example, the command is executed and forgotten immediately. The next command you enter might fail if it assumes the `CD` is still effective. Worst, it can act upon a different set of files. An `rm *` command might delete something completely different than what you intended.

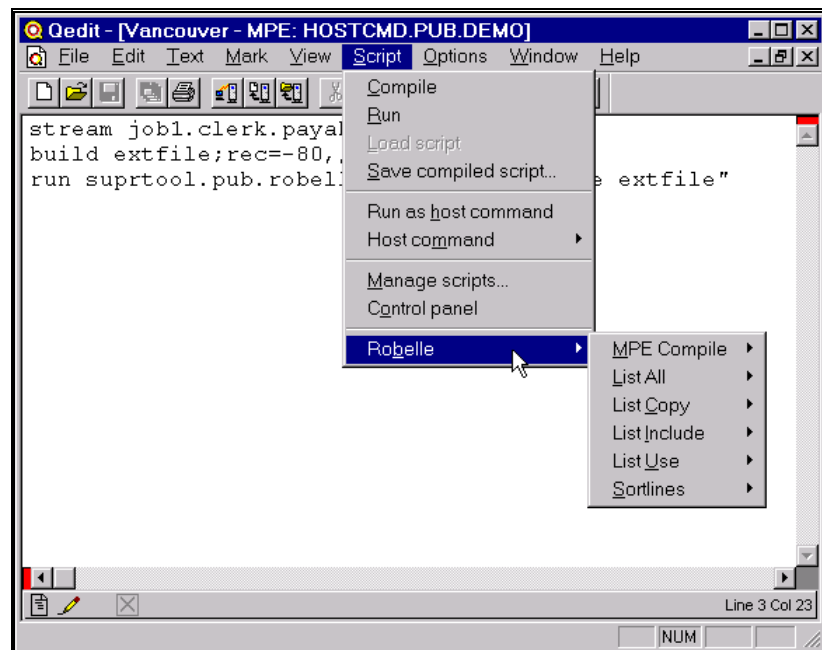
In this situation, you have 2 options. You can put all your commands in a document and use the **Run as host** command. The other option is to use the **Directory** dialog box to manually change the `CWD` to the location where you want to work.

Script-Based Features

Overview

Starting with Qedit 4.9, Qedit comes with its own scripting language. It's simply called Qedit Scripting Language (QSL). QSL allows you to extend Qedit's feature set by writing and defining functions to address your editing needs. You can also automate file editing for hands-off operations. For technical information on QSL, please refer to the *Qedit Scripting Language User Manual*.

All script-related commands are on the **Script** menu.



Script menu commands

The first set of commands, **Compile**, **Run** and **Save compiled script**, are mostly useful for script developers. The **Run as host command** command is used to execute the file as a series of host commands. The **Manage scripts** command is used to load and unload scripts. The **Control Panel** is used to display the **Script Control** dialog box. Names of loaded scripts are listed at the bottom of the menu.

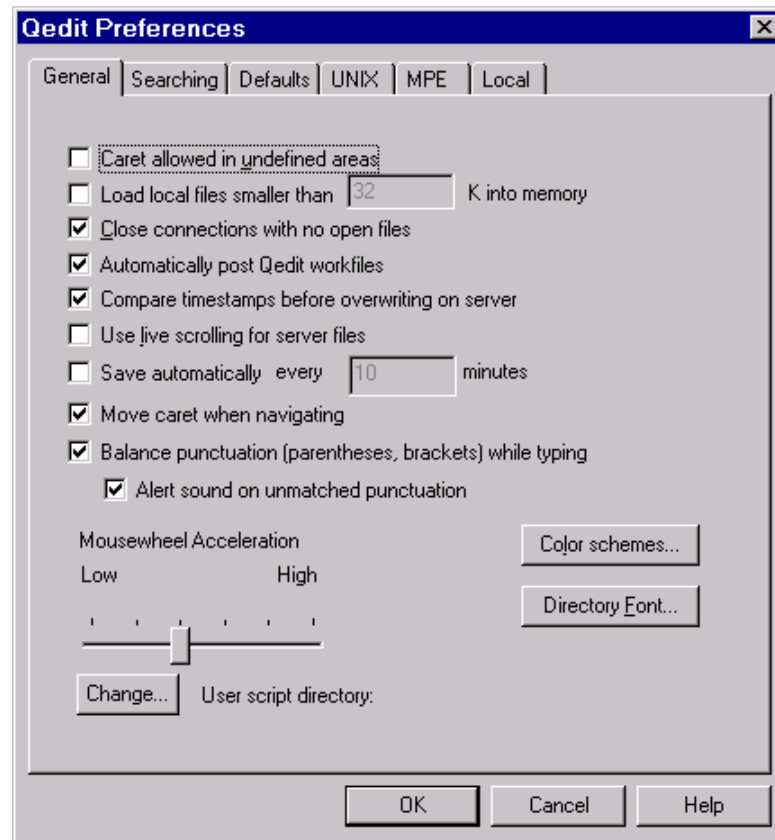
Script Libraries

QSL scripts can be stored in three different directories on your PC. The first two directories have fixed names and locations. They are created in the directory where Qedit has been installed (e.g., c:\robelle\qedit) and are called **System** and **User**, respectively.

The **System** subdirectory contains scripts provided by Robelle. This area is rebuilt every time you install a new version of Qedit. Therefore, we recommend that you do not copy any of your own files to this directory.

If you have developed scripts of your own and want to make them available to all users at your site, you should copy them to the **User** directory. This area is totally under your control.

The third location is used to store private scripts. Typically, these are scripts developed or used by a single user. The directory name and location can be different on each PC. The path name can be entered in the **User script directory** field on the **General** page of the **Preferences** dialog box.



General page of the Preferences dialog box

To specify the **User script directory**, select the **Change** button to the left of the field and browse the directory on your PC until you find the location.

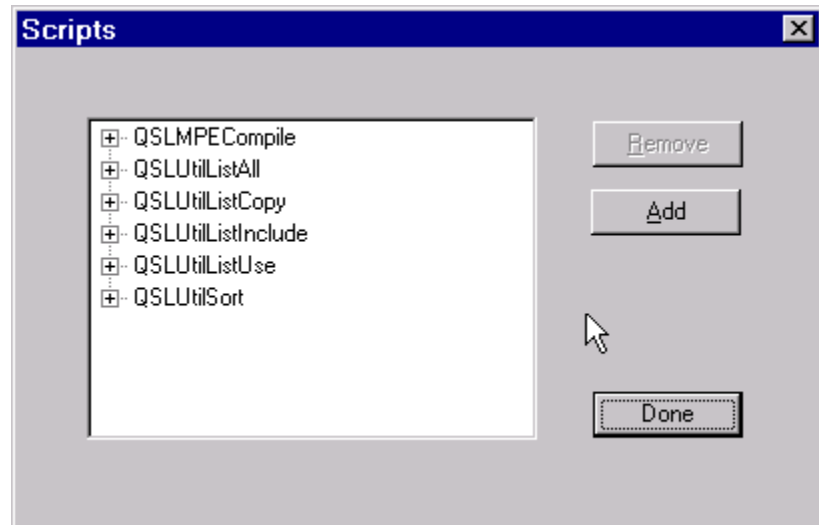
The **System**, **User** and **User Script** directories must all have three subdirectories: **Autoload**, **Scripts** and **Startup**. Each subdirectory has a specific purpose. For details, see the *Qedit Scripting Language User Manual*.

Managing Scripts

Some scripts are designed only for loading. This means they cannot be invoked directly, but can only add commands to the **Script** menu. These commands then behave like any other Qedit commands and can be used at any time.

To load a script,

1. Select the **Manage Scripts** command of the **Script** menu.



Manage Scripts dialog box

2. Select the **Add** button.
3. Browse your local directories until you find the script file you want.
4. Select the file and **Open** it.
5. The script name is automatically added to the list box on the left.
6. Select **Done**.

That's it. All the commands from that script are now available.

To remove a script:

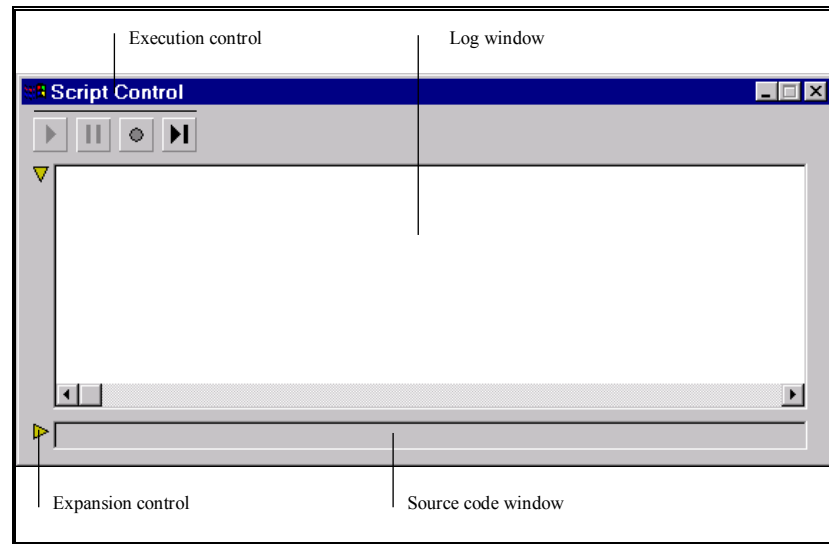
1. Select the **Manage Scripts** command in the **Script** menu.
2. Select the script name from the list box on the left.
3. Select **Remove**. The name is removed from the list.
4. Select **Done**.

For more details on the **Manage Scripts** dialog box and the Robelle script library, refer to the *Qedit Scripting Language User Manual*.

Monitoring and Controlling Scripts

Most scripts can run from the **Script** menu. However, there are times when you need additional control over their execution. A lot of scripts apply predefined edit operations and can do so without user intervention. Other scripts are designed to

display information back to the user. You can control and monitor script execution from the **Script Control** dialog box.



Script Control dialog box

The execution control buttons allow you to run, pause, stop and step through a script. The dialog box has two smaller windows which can be expanded or collapsed using the expansion control buttons (red arrowheads on the left).

Scripts can write messages to the log window. These messages can be informative for debugging purposes or simply to return the execution results.

For more details on the **Script Control** dialog box, refer to the *QSL User Manual*.

Robelle Script Library

Robelle provides a script library with each version of Qedit. These scripts are stored in the **System** subdirectory of the directory where Qedit has been installed. Other scripts can be downloaded from Robelle's Web site. This section documents Robelle-provided scripts.

Some of these scripts are automatically loaded. Others need to be loaded manually using the **Manage Scripts** dialog box. In both cases, script-based commands are available from the **Robelle** submenu of the **Script** menu.

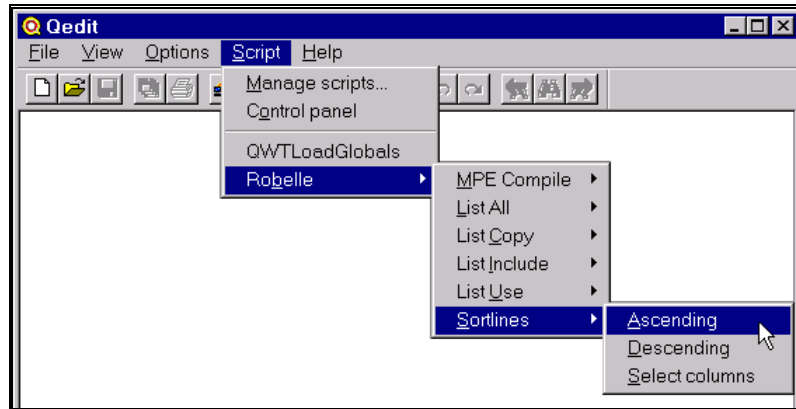
The next sections describe in more detail each script found in the library. Currently, these include:

- **SortLines**: Sorts selected text.
- **ListAll**: Lists all lines in the current document containing a specific string.
- **ListInclude**: Lists all lines containing a specific string. This script scans the current document and all Include files.
- **ListUse**: Lists all lines containing a specific string. This script scans the current document and all Use files.
- **ListCopy**: Lists all lines containing a specific string. The script scans the current document and all COBOL Copy library files.

- **MPECompile**: Compiles source files on MPE hosts.

Sorting Lines

Qedit does not have a built-in function to sort text. However, the feature has been developed using a QSL script. This script is automatically loaded on the **Robelle** submenu of the **Script** menu.



SortLines script commands

Here is how you can sort text,

1. Select all the lines that you want to sort. The script sorts all the lines that are part of the selection, even if they are only partially selected.
If you use a regular selection, Sortlines assumes you want to sort on all columns. If you use a rectangular selection, Sortlines uses the text in the selected columns as the sort keys. It re-arranges complete lines in both cases though.
2. Select the **Sortlines** command on the **Robelle** submenu of the **Script** menu.
3. If you want to sort the lines in ascending order, select the **Ascending** option. Otherwise, select the **Descending** option.

Qedit immediately starts executing the script and sorts the lines. Because the script uses Qedit's built-in edit functions, you can use the **Undo** tool to reverse the operation if you are not satisfied with the results.

Sorting on Columns

The Sortlines script sorts only the lines that are in the selection. When you try to sort on specific columns, you have to get a rectangular selection. When you have a large number of lines, it can be tedious to have to scroll through the file while holding the CTRL key to get to the last line. There are different to speed up the process.

If you wish to use the mouse,

1. Start a rectangular selection including only a few lines
2. Scroll through the file to display the last lines
3. Press and hold the SHIFT and CTRL keys
4. Click the mouse button on the rightmost sort column

This works relatively well but it requires a bit of practice to always get the columns that you want. If you are bit off when you click at step 4, the selection might not be what you want.

To make this easier, the Sortlines script provides the **Select columns** command. All you have to do is:

1. Select the columns you want to sort on a single line or make a small rectangular selection with 2 or more lines
2. Select the command on the **Script** menu

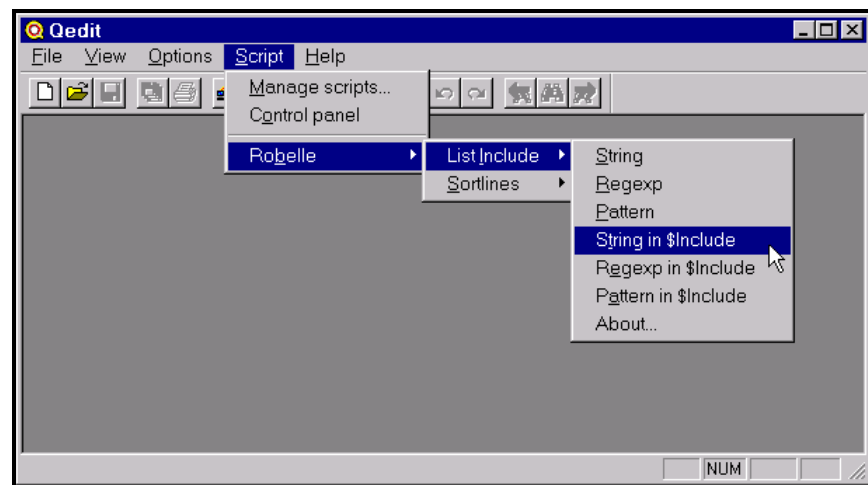
The script automatically selects the same columns on all the lines in the file.

Listing Lines

The search feature in Qedit can only find one occurrence at a time. That is, you can request Qedit to jump to the previous or next occurrence. There is no built-in function to list all the lines that contain a particular string. This feature is implemented using a series of QSL scripts:

- ListAll.qsc: Scans only the active document.
- ListInclude.qsc: Scans the active document and files referenced in \$include, !include, #include and .include statements.
- ListUse.qsc: Scans the active document and files referenced in Use statements.
- ListCopy.qsc: Scans the active document and files referenced in COBOL Copy statements.

These scripts are automatically loaded as commands on the **Robelle** submenu of the **Script** menu.



ListInclude script commands

All these scripts offer a number of options. You can search for:

- a simple string (e.g., "abcproducts.html")
- a regular expression (e.g., "[a-z0-9].html?")
- a pattern (e.g., "abc@.htm?")

All lines containing at least one occurrence of the search string are displayed in a new local file. No matter which option you use, the scripts always perform caseless searches i.e. **IgnoreCase** is TRUE. The script also displays informative messages in the log window of the **Script Control** dialog box. Here is a sample output searching for a string with the **List Include** command:

```
Searching Vancouver - MPE: COBOL4.PUB.DEMO for STRING 'error'
$Include files are searched!
Actual line numbers are displayed
Started at 07/28/2000 11:15:40 AM

Vancouver - MPE:COBOL4.PUB.DEMO: 1 line(s) found **
2.915: * Common errors include file
Vancouver - MPE:ERRMSG.PUB.DEMO: 4 line(s) found **
1      :      88 database-error    value 90.           ME000728
1.1    :      88 input-error      value 91.           ME000728
1.2    :      88 output-error     value 92.           ME000728
1.3    :      88 unknown-error    value 99.           ME000728
Vancouver - MPE:COBOL4.PUB.DEMO: 10 line(s) found **
5.4    :      perform 98-spell-error.
6.2    :      perform 98-spell-error.
7.2    :      perform 98-spell-error.
8      :      perform 98-spell-error.
8.9    :      perform 98-spell-error.
9.8    :      perform 98-spell-error.
10.7   :      perform 98-spell-error.
11.6   :      perform 98-spell-error.
11.9   :      98-spell-error      section.
12.5   :      98-spell-error-exit. goback.

Completed at 07/28/2000 11:15:40 AM
```

Searching Only the Active Document

The **List All** command searches only the active document.

Searching Referenced Files

With some compilers, it is possible to incorporate text from other files using special statements. The most common statement keyword is Include. Statements use a leading character and you can see variations like \$Include, #Include, !Include or .Include. From this point, we will just refer to them as Include files or use \$Include. Other languages have Use statements, but they work exactly the same as Include statements.

COBOL has the standard Copy statement to accomplish a similar operation. The syntax is a little bit different than Include and Use statements but the functionality is the same: reference lines from an external file.

You can use the other commands, **List Include**, **List Use** or **List Copy**, to search the active document and any referenced file it might contain. Qedit searches for appropriate reference statements based on the command you selected, opens the referenced file and then searches for the specified search string. Referenced files can be nested, which means that a referenced file can contain other reference statements. When searching for the string, Qedit scans the files as they are encountered. The output listing then shows the exact sequence as they would be interpreted by a compiler, for example.

Because there can be numerous referenced files to search, some operations can be fairly long. If this happens, Qedit displays a progress dialog box showing the file names being searched. While the server searches for the string, the barber pole moves slowly. If the server sends matched lines to the client, the barber pole moves faster. If you want to interrupt the search, click **Cancel**.



List Include(Find All) progress dialog box

Simple Search String

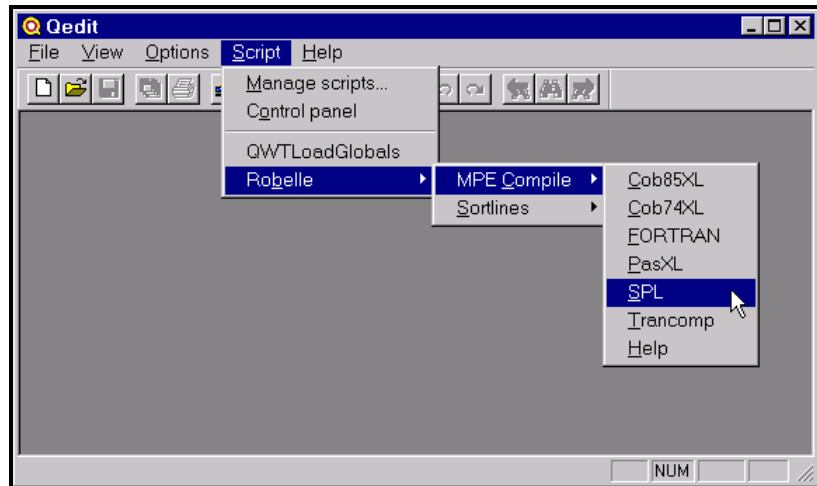
If there is no active selection, the script prompts you for one. If there is an active selection, the script uses it as a search string and starts the search immediately.

Regular Expression or Pattern Search String

If you want to search for a regular expression or a pattern, it's very unlikely that you will want to use the selection. For this reason, the script is always going to prompt you for a search string. If there is an active selection, it is going to be used as the default value, but you will be able to change it before the search operation starts.

Compiling MPE Source Files

If you are writing programs on an MPE host, you often need to compile them. You can use the host command dialog box to invoke the compiler of your choice. However, this is sometimes tedious to move back and forth between the dialog box and the source document. In the Robelle script library, we provide a simple script that gives easy access to most compilers available on MPE. This script can be loaded from the **Scripts** subdirectory of the **System** directory using the **Manage Scripts** command of the **Script** menu. Loading this script adds commands to the **Robelle** submenu of the **Script** menu.



MPECompile script commands

The script invokes the appropriate compiler on the active document. It supports the Cob85XL and Cob74XL commands for COBOL programs, the Fortran command, the PasXL command for Pascal programs and the Trancomp command for TRANSACT programs. The script simply invokes the compiler with the source file name and redirects the output to a temporary file called QSLCOMPO. For example, a COBOL compile for COBPRG1 would be:

```
Cob85x1 COBPRG1.SRC.DEVACCT, ,QSLCOMPO
```

The compiler output file is then opened in a new document window.

Power Tips for Developers

Overview

If you're looking for ways to use Qedit as efficiently as possible in a programming environment, here are some tips for you.

Spanning Hosts and Operating Systems

Whether you need to access files on MPE, UNIX or Windows, Qedit lets you work on all of these operating systems from a single application. What's even better is that you can work on any of these hosts, or any of your local files, all at the same time. That means you can conveniently and seamlessly edit files (see "Typing, Selecting & Editing Text" on page 87), as well as move or copy them across platforms (see "Copying & Printing Files" on page 147).

Switching and Tiling Files

With access to multiple document windows, you can easily switch between different Qedit files or you can switch to another application. See "Switching Between Applications" on page 37. For example, you can have source code in one window and its data in another. Or compare the shell script that is working properly on one host with the same script that is failing on another host. Qedit lets you work and maneuver through all your files uninterrupted.

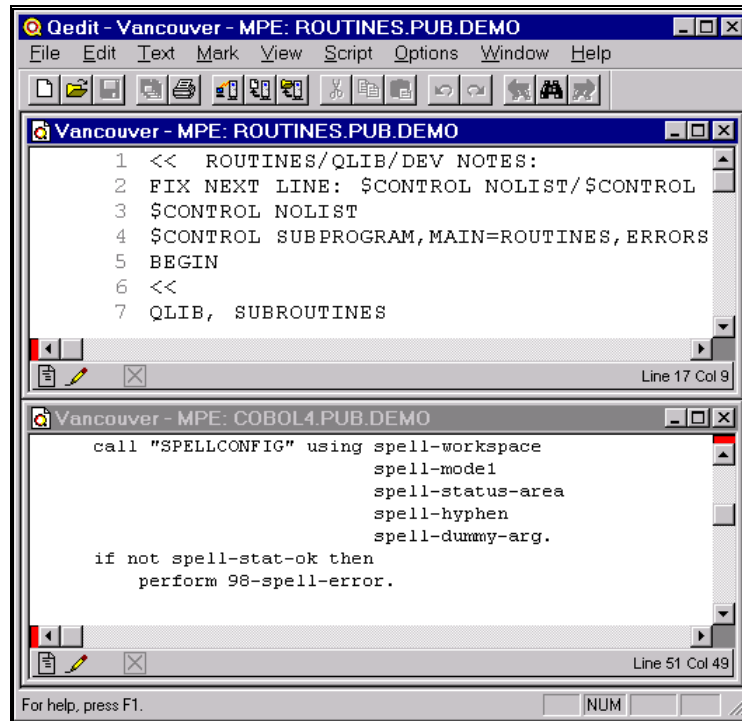
Compare different parts of the same file by splitting a document window into multiple views. See "Splitting a Document into Independent Views" on page 35. And if you want to look at different files side by side, simply tile all your open files or only those you need.

Switching Between Files

To switch between files use CTRL+TAB. Every time you press TAB as you hold the CTRL key, you will switch to the next open file. Or, switch to the file you want by choosing one from the **Window** menu, where all open files are listed in numbered order (e.g., ALT+W,1 takes you to the first file).

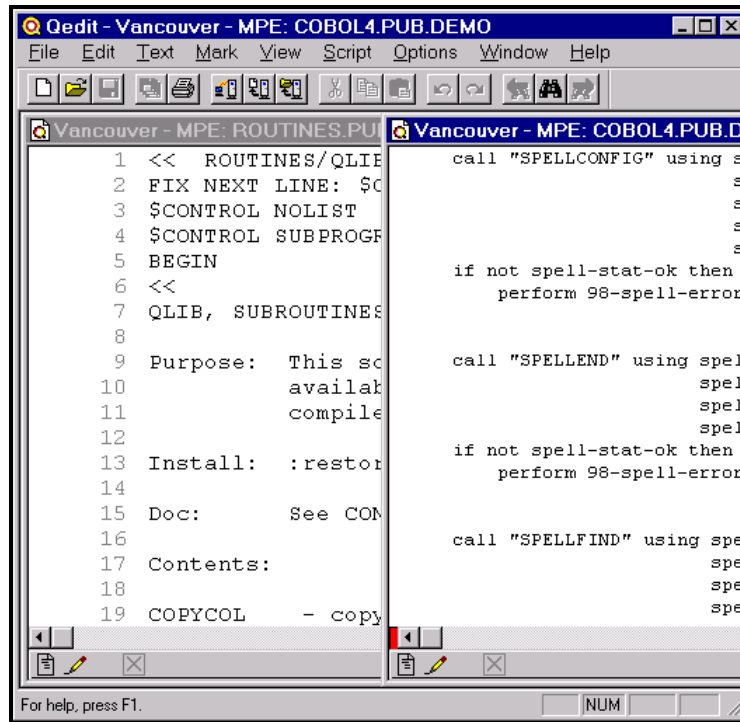
Tiling Document Windows

Tile two documents horizontally by typing ALT+W, H. This will put currently opened windows one on top of the other. You will now see something like the picture below. Qedit uses the full width of the application workspace and fits as many windows as possible in the available height.



Horizontally tiled windows with two documents

If you prefer to see the windows side by side, use ALT+W, V. The application workspace will now look something like this. It uses the full height of the application workspace. Qedit tries to fit as many windows as possible in the width available.



Vertically tiled windows with two documents

You can switch between documents quickly by just moving your mouse to another document.

What if you have four open documents and the auto-tiling format isn't what you want (i.e., you want to look at two at a time, but in pairs)? You can easily adjust the windows manually. Minimize the two you don't need to look at for the moment. Now tile the remaining two documents with ALT+W,H. Suppose you want the upper window to occupy more space. Drag the top border of the lower window down to make the lower window smaller. Now just drag the bottom edge of the upper window down and make it larger!

What if you want both windows to be larger, leaving a bit of the other window visible when they are active. That's easy, because windows can overlap. Use ALT+W,H to tile the windows, minimize those you don't want, then use ALT+W,H again. Now drag the bottom edge of the lower window as far down as it will go and drag the top edge up, leaving just a portion of the other window visible. Finally, click the upper window and drag the bottom edge down, leaving just a portion of the lower window visible. Now when you switch windows, the newly active window will always expand, while still leaving part of the other window visible to scroll and click.

Now here is a test. Configure your four windows so that two of them are located at the top and cover 3/4 of the Qedit workspace, and the other two are located at the bottom and come up 3/4 of the way. Hint: adjust the location and size of each independently, then use CTRL+TAB to switch between them.

Working Smarter with a Terminal Window

Open up a terminal window on your host using a tool such as Hyper Terminal or Reflection. You will need a terminal window to test an application whose source files you are modifying with Qedit for Windows. Now you can switch between the edit windows and an execution window.

Consider some typical tasks. To compile your source file, you need to ensure that your local edits are posted to the host server, then execute a compiler program on the server. If you are editing a regular file, you will want to use the **Save** command in the **File** menu before going to the terminal window and compiling the file. Otherwise you will be compiling the old version. If you are editing a named Qedit workfile, you do not have to save your changes because when you leave Qedit for Windows, it automatically posts changes to all open Qedit workfiles on all hosts. If you find compiling errors, just click Qedit for Windows and use the **Goto** command in the **Mark** menu to jump to the line with the syntax error.

To execute your revised program, just stay in the terminal window after a successful compile, link the program and type the command to execute it.

Advantages of Qedit Workfiles

Automatic Save

Changes to a Qedit workfile are posted to the server immediately. You do not need to save your work explicitly. See "Saving a File" on page 157.

Sole Access to Files

If you want to make sure that only one person at a time has access to a file, you can convert a file to the Qedit workfile format. This still gives a group of people access to the file, but only the first person to open the file gets write access. Storing a file in a Qedit workfile format ensures that no one else can modify the file while you have it open. See "Converting a File to Qedit Format" on page 150.

Preventing Accidental Overwrite of Files

If you are not working on Qedit workfiles, there can be simultaneous access to file. To prevent another user from changing a file while you are working on a copy of it, you can enable a feature that warns you about a possible overwrite. This gives you the ability to decide if someone else's changes can be overwritten when you save your changes. See "Comparing Timestamps Before Overwriting Host Files" on page 162.

Qedit server 5.1.10 or later is required.

If you wish to get tighter control over host files you are working on, you can enable the Set Text Exclusive feature of the server. You need to enter the command in one of the server configuration files (Qeditmgr). For this feature to be most efficient, we recommend that you use one of the system configuration files. For complete details, refer to the appropriate *Qedit User Manual*.

If this feature is enabled, users will not be able to open a regular file if someone else is already working on it. If a user doesn't have the feature enabled, he will be able to open the file but will not be able to save his changes until the file is released.

Instant Access to Embedded Files

If you are viewing a file in Qedit and come across an embedded file name, one of the quickest ways to open the file is to highlight the name and press CTRL+D. This feature is especially useful with Include files. If the embedded file is a host file that needs a connection, Qedit has a systematic approach to select a connection. See "Opening a File from a Command Line" on page 73 and "Opening Embedded Files" on page 74.

Getting the Fully Qualified Name of a File

If you want to include the fully qualified name of your current file in another document or an electronic mail message, simply do the following:

ALT+F, P

1. Open the source file and select **Properties** from the **File** menu.
2. When the **Properties** dialog box appears, click the **Information** tab.
3. Select **Copy** to put the contents of the **Information** tab on the Clipboard. To isolate only the file name, paste all the file information into a temporary file, delete everything except the file name, then copy the file name to the Clipboard.
4. Go to the target file or application, and paste the file name from the Clipboard.

That's the long way to do it. You can also use the **Copy Pathname to Clipboard** command on the shortcut menu (right-click).

Handling Nonprinting Characters

If you are working on files with nonprinting characters, Qedit preserves these characters when you edit the file. And if you need to edit the nonprinting characters, you can set up a file library of nonprinting characters and use Qedit's convenient cut-and-paste feature to edit them. To set up a file library of nonprinting characters, see "Inserting Nonprinting Characters" on page 90.

Minimizing Keystrokes

If you prefer to use a keyboard, Qedit's many shortcuts minimize your keystrokes. In addition to common Windows editing functions, Qedit shortcuts include searching for and moving text as well as creating and opening files.

Menu Shortcuts

As you grow more familiar with Qedit's features, you can incorporate keystroke shortcuts to quickly access application menus, dialog boxes, editing and searching features. It takes only a few keystrokes to put you into vital parts of the program. See "Key Combinations and Functions" on page 200.

Letter Matching

You can select a file you previously opened by typing only the first few letters of its name in a text box. Qedit matches the letters you type with the file names in its list.

By carefully choosing unique and meaningful file and connection names, file selection becomes faster and more efficient.

Expanding the Workspace

If you want Qedit to display a few extra lines, you can turn the application toolbar into a floating toolbox and move it to another part of the window. This will give you more lines per screen and a larger working area. For more information, see “Turning the Toolbar into a Floating Toolbox” on page 27.

Quick Access to Directories

If you're not sure of a file's name and you want to browse through a host directory to find it, here's a quick way to get a directory listing of your current working directory or group: click and hold the **Manage Connections** button on the toolbar. When a shortcut menu appears, select the saved connection for which you want to get the directory listing. As soon as you choose a connection, the **Directory** dialog box will appear and you can browse through the listing.

Working with MPE Hosts

This section contains information that pertains strictly to MPE hosts.

Working with Copylibs

On HP e3000 computers, COBOL Copy modules are stored in KSAM files. These library files are called Copylibs. Qedit for Windows can open modules that reside in the file with the same name as your logon group. If the Copylib you use is not in your logon group, you can have a file equation in the `Qeditmgr.Pub.Sys` file that points to the file's current location. Qedit/server will execute commands in the file `Qeditmgr.Pub.Sys`, which it does for everyone connecting to the host.

It might be a good idea to use an `If` command to select your personal preferences. This example checks the session name and issues the file `equate` if the name contains the string “george.” Of course, you have to put your session's name in the logon string for that connection.

```
If hpsession = "george" then
    file copylib=mylib.srcinc.acct
endif
```

To open a Copylib module, specify the name in parentheses. For example, you can open the “vmisc” module by specifying it as the file name in the **Open Host File** dialog box.

You can get a list of all the modules in the Copylib by entering an at sign in parentheses (@) as the file name. Member names are listed in a new document window.

User Labels on Files

The MPE file system stores file characteristics in a special area called file labels. These labels are not part of the data and are maintained by the operating system. To

extend this feature, MPE allows the creation of user labels. An application can store additional information about the file or its contents. Because user labels are under application control, they can be erased or modified.

The MPE server is able to detect user labels but, by default, does not preserve them. For example, if you open such a file and save it back, the user labels are removed. If you want to preserve user labels, if any, you have to add the following line to one of your server configuration files:

```
Set Work Labels On
```

This does not affect editing operations in any way.

Network Issues

TCP/IP

Qedit works best on a true TCP/IP connection created with a Winsock 1.1 compliant TCP/IP stack.

Qedit should work with proxy servers. This is often used in corporate intranets to allow the people on the LAN to surf the net without exposing their actual IP addresses to the whole Internet. When using a proxy server, you will likely have to use Qedit's **Firewall protocol** on MPE connections. In some installations, you might have to configure the proxy server to ignore the host's IP address.

Firewalls

If you want to use Qedit to access corporate servers from home or a remote office, you might have to reconfigure your firewall software or router. With the **Firewall protocol**, the connection to all UNIX and MPE hosts occurs on port 7395.

Connections to MPE hosts start off on port 1542, which is the default, and continue on other ports. If you have to use Qedit's firewall protocol, you might have to change your firewall with the appropriate port number. See "MPE Host Connections" on page 211.

Most firewall configurations are done by IP address. If you use an ISP when you connect from home, you might be assigned a random IP address, which can make configuring the firewall more difficult. Most ISPs, however, use a small range of addresses for a specific user.

Client to Server Handshaking

In order to detect and report possible connection problems during a session, the client and the server programs are constantly exchanging information. Of course, if you are working on a host file, information is always flowing: your changes are transmitted to the server and the server returns lines as you scroll through the file. Every time the client sends a transaction over the network, the server sends back a transaction to acknowledge receiving the client's request.

If you have started a connection but are not actively using it (you might be working on a different host or have simply left your office for a moment), the client keeps

sending special transactions to the server to make sure the connection is still functional. These transactions are called **KeepAlive**. If you enable tracing on the server and look at the trace file, you would see something like this:

```
Sep 22, 1997 09:31:39 [187] TRead. Length=12,KeepAlive,version=1,id=0
Sep 22, 1997 09:31:39 [187] TSent. Length=12,KeepAlive,version=1,id=0
```

The first entry identifies the KeepAlive transaction received from the client. The second entry shows the server sending the acknowledgment back.

The client sends such a transaction every minute and expects an answer from the server almost immediately. If the client does not receive an answer from the server after 3 KeepAlive transactions (i.e., after about 3 minutes), the client assumes there is a problem and disconnects.

The server does not send any transactions of its own. It only responds to client requests. If there is a legitimate problem with the network and the client cannot communicate with the server anymore, how could the server know there is a problem?

On MPE hosts, there is also a built-in timeout value. If the server does not receive a transaction of any kind within 5 minutes, it assumes the client is no longer there and breaks the connection on the host side. This means dead sessions will eventually go away by themselves. However, if the client aborts and you were using a Qedit workfile or had a scratch file opened, you will not be able to re-open them during that 5-minute period. Qedit will report that the file is currently accessed by someone. You simply have to wait for the timeout to expire.

On HP-UX hosts, there is also a timeout function and server processes should disappear by themselves. However, in the event that some old server processes remain on your system, you can get rid of them by logging in as root on the system console or on a terminal and entering `ps -ef | grep qedit`.

This command looks at all the processes currently running on your system and selects only the ones that contain the string “qedit.” You should get a listing similar to this:

```
root 16554 24018 2 12:05:29 ttyp4 0:00 grep qedit
joe 14043 12235 0 09:16:57 ? 0:10 /opt/robelle/bin/qedit -d -t
root 12235 1 0 Oct 15 ? 0:00 /opt/robelle/bin/qedit -d -t
cybil 5449 12235 0 06:44:22 ? 0:01 /opt/robelle/bin/qedit -d -t
ricky 15948 12235 0 Oct 10 ? 0:01 /opt/robelle/bin/qedit -d -t
```

The last column on the right shows the name of the programs selected. The line that reads “grep qedit” is the command you used to get this listing. Ignore it. All the other ones (running `/opt/robelle/bin/qedit -d -t`) are Qedit server instances.

The line that shows a one (1) in the third column is actually the Qedit listener process. This process will be constantly running until you decide to shut it down. You would do so if you wanted to make sure no one can connect to this host using Qedit. For example, you might stop the main Qedit process while you are backing up the system. You also have to remove this process if you want to install a new version of Qedit for HP-UX.

The other three Qedit processes (joe, cybil and ricky) are actual Qedit connections. The fifth column indicates the time that process was started. If you see only a time (e.g., 09:16:57), this means the process has been running for less than 24 hours. It is the actual time it started. If the process has been running for more than 24 hours, the system shows only the date it started (e.g., Oct 10).

Based on this information and your knowledge of your system’s normal usage, you can decide which processes are valid. In this case, ricky’s process is suspect because it started before the main listener process, Oct 15. Invalid or ghost processes can be

removed with the UNIX kill command. For example, if you want to get rid of ricky's process, enter the command followed by the process id number showing in the second column.

```
kill 15948
```

Looking at Server Connections

From a system management point of view, it is sometimes useful to find current connections to a host. Both MPE and UNIX hosts have the information available through commands or utilities.

MPE Connection List

Users require special capabilities to run nettool.

The `sockinfo` command of the `nettool` program displays information on sockets currently used on the host. The complete program name is `nettool.net.sys`.

```
:nettool.net.sys
NETTOOL Version:      B06000A1

Enter HELP HELP for an explanation of the help facility.

Enter HELP OPERATION for guidance on NETTOOL use.

Commands can be abbreviated, HELP keywords cannot.

Menu-specific commands for the ROOT:

RESOURCE   CONFIG      NAMEADDR    PING        STATUS      NMDUMP
X25STAT    X25CHECK    IPCINT      XPVAL       XPPERF     QVALNS
NSTEST     LOOPINIT    NSLOGON     SOCKINFO    filters

[1]>>>
```

At the `nettool` prompt, you then enter the `sockinfo` command. The program automatically goes to the **Global Display** where all currently opened sockets are listed. On a busy system, the list can be quite long. The following example has been edited and shows the format.

```
[1]>>>sockinfo
GLOBAL DISPLAY                               Gsxds=$b.86263000    12:04 pm

Pin  User                Program                Job      Pri  Skts
-----
122* ricky,clerk.acctg  qedit.pub.robelle     s3582   cq152  1
124* jinetd,manager.sys inetd.net.sys          j5926   cq152  2
127* jhttpd,mgr.apache httpd.pub.apache       j5905   de202  1
131* dan,pgrm.dev     qedit.pub.robelle     s3583   cq152  1
138* (system process) snmp.net.sys           s3583   cq152  2
154* (system process) dsdad.net.sys         dsdad   lq149  12
163* dan,pgrm.dev     vtserver.net.sys      s3583   cq152  1
-----
Totals: 28 processes, 0 locked semaphores; 40 sockets.
```

To identify Qedit connections, you would look for the server name currently configured in the client in the **Program** column. By default, the server is `qedit.pub.robelle`. If you know the logon information of a particular user, you would look for it in the **User** column. The other columns that might of interest are **Pin** and **Job**. The process identification number (PIN) is the server instance associated to a particular connection. This number should match the **Host Process ID** line found in the **Information** page of the file **Properties** dialog box.

You can use the `Showproc` command to get details on how the server has been invoked. The server name is displayed along with the encoded IP address and socket number.

```
:showproc 131
QPRI CPUTIME STATE JOBNUM PIN (PROGRAM) STEP
C152 0:00.708 WAIT S3583 131 (QEDIT.PUB.ROBELLE) -dxP18c915cb052a -$
```

UNIX Connection List

On UNIX hosts, you can find information about server instances using the `ps -ef` command. This command produces a list of all processes currently running on the system. To get a list of Qedit server processes, simply pipe the output from `ps` to `grep` as in:

```
ps -ef | grep qedit
```

The output from this command would look something like:

```
root 16554 24018 2 12:05:29 ttyp4 0:00 grep qedit
joe 14043 12235 0 09:16:57 ? 0:10 /opt/robelle/bin/qedit -d -t
root 12235 1 0 Oct 15 ? 0:00 /opt/robelle/bin/qedit -d -t
cybil 5449 12235 0 06:44:22 ? 0:01 /opt/robelle/bin/qedit -d -t
ricky 15948 12235 0 Oct 10 ? 0:01 /opt/robelle/bin/qedit -d -t
```

To display information on sockets, you can use the `netstat` command. Here is an edited example of a `netstat` listing:

```
$ netstat
Active Internet connections
Proto Recv-Q Send-Q Local Address Foreign Address (state)
tcp 0 0 produx.4279 produx.domain TIME_WAIT
tcp 0 0 produx.domain produx.4277 TIME_WAIT
tcp 0 0 produx.http 22.33.44.55.0r.1126 TIME_WAIT
tcp 0 0 produx.7395 22.56.77.199.1660 ESTABLISHED
```

The **Local Address** column shows the host name and port number which sockets are using. In the example above, the host name is `produx` and the port number is `7395` (Qedit default). The **Foreign Address** column provides information on the source workstation. Typically, this is the IP address of the client PC running Qedit.

The listing can be quite lengthy. If you wish to get only information pertaining to Qedit, you could pipe the `netstat` output into `grep` as in:

```
netstat | grep 7395
```

Reference

Menu Commands

File Menu

The File menu on the toolbar contains the following commands: New, (New) Host, (New) Local, Open, (Open) Host, (Open) Local, (Open) Open Selection, Recent Files, Favorites, Revert, Close, Disconnect, Save, Save As, Save All, Properties, Print, Page Setup, Print on Host, and Exit.

New

Choosing New creates a new file. The submenu appears offering two choices: create a new Host file or a new Local file.

(New) Host

Choosing (New) Host prompts the New Host File dialog box. This is where you establish the Connection, Record Length, and Language of the file.

(New) Local

Selecting (New) Local creates a new local file automatically.

Open

Selecting Open prompts the (Open) Host and (Open) Local file submenu. If a file name is highlighted, it also prompts for the (Open) Open Selection submenu.

(Open) Host

Choosing (Open) Host prompts the Open Host dialog box.

(Open) Local

Choosing (Open) Local prompts the Open dialog box.

(Open) Open Selection

If a file is already opened, Open Selection also appears in the Open submenu. This command allows you to open another file whose highlighted name appears within the open file.

Recent Files

Lists the most recently used files.

Favorites

Lists the user's favorites files and subfolders.

Revert

Prompts the Confirm Revert dialog box.

Close

Closes the current open file.

Disconnect

Closes a connection between the client and host.

Save

Saves any changes since the last save operation.

Save As

Used to rename a file or change the file name, type, directory, or path.

Save All

Saves changes to all open files on Qedit.

Properties

Prompts the file Properties dialog box. The dialog box provides tabbed pages to set general file Options, display and set COBX tags, display file-related Information, change Tabs and Guides settings.

Print

Prompts the Print dialog box in which you choose print settings.

Page Setup

Prompts the Page Setup dialog box in which you choose layout options.

Print on Host

Prompts the Print on Host dialog box in which you select the print destination. This command is not available for local files.

Exit

Quits the program, closes all files and disconnects from all hosts.

Edit Menu

The Edit menu on the toolbar contains the following commands: Undo/Can't Undo, Redo/Can't Redo, Cut, Copy, Paste, Copy Special, Clear, Select all, Find, Find again, Find again reverse, Find previous, Find next, Find selection, Find previous selection, Replace and Column.

Undo/Can't Undo

Reverses the last action, unless the action cannot be undone.

Redo/Can't Redo

Lets you re-apply the last undone operation, unless it cannot be redone.

Cut

Removes a highlighted selection to the Clipboard so that it can be moved. The Clipboard can only retain one cut or copied selection at a time. See Copy.

Copy

Lets you copy a highlighted selection to the Clipboard so that it can be moved. The Clipboard can only retain one cut or copied selection at a time. See Cut.

Paste

Retrieves the cut or copied selection from the Clipboard.

Copy Special

Prompts the Copy Special dialog box which allows control over special Copy operations to the local Clipboard.

Clear

Removes a highlighted selection and does not post it to the Clipboard. This is equivalent to pressing the DELETE key. If the selection is rectangular, Qedit replaces the contents of the selection with spaces.

Select all

Highlights the contents of the entire file.

Find

Prompts the Find String dialog box to search for a string, starting from the current cursor position.

Find again

Repeats a search for a string in a document, starting from the last string found.

Find again reverse

Reverses the direction of the search for the same string, starting from the current cursor position.

Find previous

Searches for the previous occurrence of the string, starting from the current cursor position.

Find next

Searches for the next occurrence of the string, starting from the current cursor position.

Find selection

Searches for a selection highlighted in the file, starting from the highlighted text position.

Find previous selection

Searches for the previous occurrence of the highlighted selection, starting from the highlighted text position.

Replace

Searches for a string and replaces it.

Column

Provides access to column edit commands.

(Column) Delete

Removes text from one or more columns. Optionally, replace the text with spaces.

(Column) Insert

Inserts text in a specific column.

(Column) Shift

Shifts text in certain columns, left or right.

Text Menu

Detab

Replaces tab characters with appropriate number of spaces.

Entab

Replaces spaces with tab characters where appropriate.

Shift left

Shifts the line left to the previous tab stop.

Shift right

Shifts the line right to the next tab stop.

Mark Menu

The Mark menu on the toolbar contains the following commands: Save position, Go to previous position, Exchange position, and Goto line.

Save position

Marks up to ten positions in a file so that you can return to each of them.

Go to previous position

Returns to markers in Last In/First Out order.

Exchange position

Switches your current position with your last marked position. You can then return to the original position.

Goto line

Move to a line in a file.

View Menu

The View menu on the toolbar contains the following commands: Ruler bar, Record numbers, Toolbar, Status bar, and Show invisibles.

Ruler bar

Displays a horizontal ruler for tabs and guidelines.

Record numbers

Displays record numbers in a file.

Toolbar

Displays the Qedit toolbar.

Status Bar

Displays the status of Qedit's toggle keys.

Show invisibles

Displays formatting characters that are normally invisible.

Script Menu

The Script menu on the toolbar contains the following commands: Compile, Run, Load script, Save compiled script, Run as host, Host Command, Manage scripts and Control panel. Other commands on the menu are custom commands and added by previously loaded scripts.

Compile

Compiles the script stored in the active document.

Run

Compiles and runs the script stored in the active document.

Load script

Loads a script in the Qedit script environment.

Save compiled script

Compiles the script stored in the active document and saves the result in compiled format.

Run as host command

Executes the file contents or lines in the selection as host commands.

Host Command

Prompts the host command dialog box in which you can execute commands on a specific connection.

Manage scripts

Prompts the Manage scripts dialog box which allows loading and unloading of scripts.

Control panel

Prompts the Script Control dialog box which allows control over script execution, display script messages and source code for debugging purposes.

Options Menu

The Options menu on the toolbar contains the following commands: Preferences, Connections, Debug, and Server.

Preferences

Prompts the Qedit Preferences dialog box in which you choose default settings for General, Searching and Defaults options.

Connections

Prompts the Connection List dialog box in which you Add, Remove, and view Details of the connection.

Debug

Prompts the Debug Controls dialog box in which you set tracing features.

Server

Prompts the Configure Server Settings dialog box.

Window Menu

The Window menu on the toolbar contains the following commands: Cascade, Tile Horizontal, Tile Vertical, Arrange icons, Split Window, (Split Window) Horizontal, (Split Window) Vertical, and a list of currently opened files.

Cascade

Arranges all open files in a stacking sequence from the top left corner of the screen.

Tile horizontal

Fills the application window with all the open documents. Document windows are placed one on top of the other.

Tile vertical

Fills the application window with all the open documents. Document windows are placed side by side.

Arrange icons

Arranges minimized document windows into a neat row, side by side.

Split Window

Displays a submenu with two commands, Horizontal and Vertical, which split the active window or pane in half in either direction.

(Split Window) Horizontal

Splits the active window horizontally in half.

(Split Window) Vertical

Splits the active window vertically in half.

Help Menu

Provides on-line help for Qedit for Windows. The Help menu on the toolbar contains the following commands: Tip of the Day, Contents, Using Help, Scripting, Robelle Web, (Robelle Web) Robelle Web Site, (Robelle Web) What's New, (Robelle Web) Scripting, (Robelle Web) E-Mail Technical Support, and About Qedit.

Tip of the Day

Prompts the Tip of the Day dialog box.

Contents

Accesses the application on-line help through the Help Topics dialog box.

Using Help

Accesses information about using on-line help.

Robelle Web

Provides easy access to Robelle resources over the Internet.

(Robelle Web) Robelle Web Site

Invokes the default Web browser and connects to Robelle's main page.

(Robelle Web) What's New

Invokes the default Web browser and connects to the Qedit for Windows What's New page.

(Robelle Web) Scripting

Invokes the default Web browser and connects to the Qedit for Windows Scripting page.

(Robelle Web) E-mail Technical Support

Allows you send an e-mail message to Robelle's technical support staff (support@robelle.com).

About Qedit

Prompts the About Qedit for Windows dialog box.

Key Combinations and Functions

Control Combination Keys

<u>To Move:</u>	<u>Use:</u>
Up in a file in the same line and column, line by line.	CTRL+↑
Down in a file in the same line and column, line by line.	CTRL+↓
To the preceding "start of word."	CTRL+←
To the next following "start of word."	CTRL+→
To the first column of the first line in a file.	CTRL+HOME
To the last character in the last line of a file.	CTRL+END
One page up in a document window, less one line.	CTRL+PAGE UP
One page down in a document window, less one line	CTRL+PAGE DOWN

Searching Shortcuts

<u>To Find:</u>	<u>Use:</u>
A new string in a file from your current cursor position.	CTRL+F

The next occurrence of a string with the same options as the last search.	CTRL+G
The next occurrence of a string in the opposite direction of the last search.	CTRL+SHIFT+G
The next occurrence of a string towards the end of the file.	CTRL+B
The next occurrence of highlighted text from the current cursor position.	CTRL+H
The previous occurrence of the highlighted text from the position of the last selection found.	CTRL+SHIFT+H
The previous occurrence of a string towards the beginning of a file from the last cursor position.	CTRL+U
A string and replace it with a another string.	CTRL+R
A specific line or record number in the current file.	CTRL+J

Editing Shortcuts

Copy selected text to the Clipboard.	CTRL+C
Paste text from the Clipboard to the current cursor position.	CTRL+V
Cut selected text to the Clipboard.	CTRL+X
Select all text in a file.	CTRL+A
Undo last operation	CTRL+Z
Save changes made to file.	CTRL+S

Other Shortcuts

Create a new local file.	CTRL+N
Create a new host file.	CTRL+SHIFT+N
Open an existing local file.	CTRL+O
Open an existing host file.	CTRL+SHIFT+N
Open the file name highlighted in a file.	CTRL+D
Print the current file on a local printer.	CTRL+P
Switch between document windows.	CTRL+TAB
Switch to the previous document window.	CTRL+SHIFT+TAB
Switch between all active applications.	ALT+TAB
Display the document control menu.	ALT+HYPHEN
Display the application control menu.	ALT+SPACEBAR

Function Keys

Display on-line Help.	F1
Move the cursor up toward the beginning of a file.	F2
Find the previous occurrence of a string.	F3
Find the next occurrence of a string.	F4
Go to the previous page.	F5
Go to the next page.	F6
Mark the current cursor position.	F11
Return to the last marked position.	F12
Close the current document window.	CTRL+F4
Switch to the next document window.	CTRL+F6
Exchange the current cursor position with the last marked position.	CTRL+F11
Select text between the current cursor position and the last marked position	SHIFT+F12
Close the program and all open files.	ALT+F4

Troubleshooting

Overview

This chapter describes the information sources to track a problem as well as an overview of some potential areas of trouble: connecting to hosts, server problems and recovering files. As users need to access multiple computers, the issue of setting and synchronizing time zones is also discussed.

What to do in case of problems?

If you are experiencing some problems with Qedit, you can call our technical support team. In order to get a better response and faster resolution, there are certain steps you can go through before calling. These steps allow you to have as much information as possible available for the technical staff.

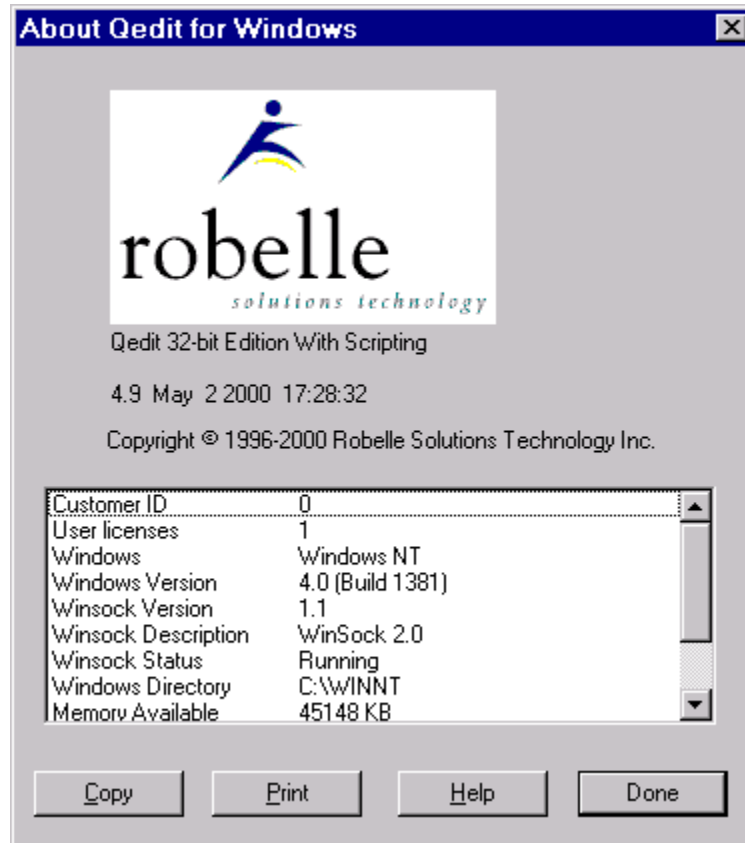
There are basic questions you should ask yourself:

- Is the problem reproducible?
- Is it happening on one PC in particular?
- Is it happening with one server or connection in particular?
- Has it worked before? If so, when did it start?

You should also gather information on the equipment you are using:

- What type of PC are you using?
- What type of connection are you using (modem, LAN, Internet, etc.)?
- What type of server are you connecting to (MPE, HP-UX, Windows NT, etc.)?
- What versions of the operating system(s) are you using?

Qedit provides some tools to answer some of these questions. You can use the **About Qedit** command of the **Help** menu to see part of your PC configuration.



About Qedit dialog box

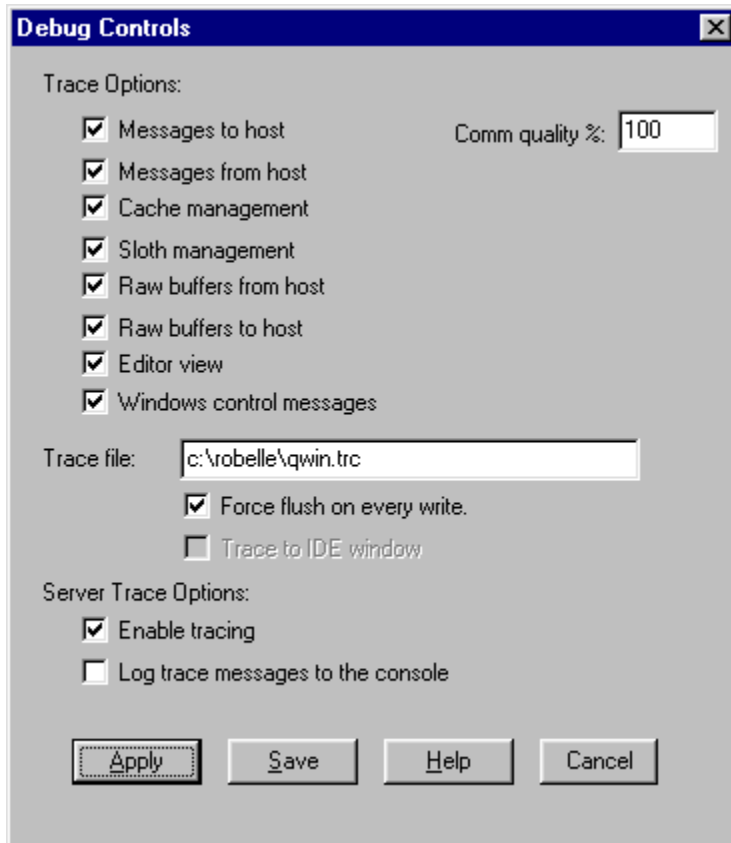
This dialog box shows not only Qedit's own version number and build timestamp but also information related to the Qedit license, your Windows environment, Windows socket (TCP/IP stack) used, memory usage and display resolution.

You can select **Copy** to send this information to the Clipboard. You will then be able to paste it back into another application such as electronic mail software. If you prefer, you can get a hard copy by selecting **Print**.

If you are able to connect to a host but are having difficulty editing host files, you can get more information by using the **Properties** command in the **File** menu and displaying the **Information** tab.

Getting a Trace

If the problem cannot be resolved easily, technical support might request you to get trace information from Qedit. This can be done by using the **Debug** command in the **Options** menu. This command displays the **Debug Controls** dialog box.



Debug Controls dialog box

The trace options on the dialog box can be divided into two categories: local trace options and server trace options. The first option simply enables the trace feature on the server. The second option forces the server to write trace records to the host console as well as the trace file on disk. In most cases, when you have to use trace, it is better to enable all the options available for the local trace:

- Messages to host
- Messages from host
- Cache management
- Sloth management
- Raw buffers from host
- Raw buffers to host
- Editor view
- Windows control messages



Warning

Trace files can become very large, very quickly especially when working with large data files. We recommend that you leave the trace option disabled until you have a need for it.

The **Comm Quality %** text box should always specify 100. You have to give the trace file a name. It does not have to be in any particular directory, but we

recommend you put it in the same directory as the Qedit program. We also recommend that you use a `.trc` extension. Enable **Force flush on every write** to ensure all the trace records are posted to disk immediately.

Server Trace Files

When **Enable Tracing** of the **Server Trace Options** is checked, the server writes information to three (3) server trace files. On MPE hosts, the files are created in the Qlog group in the account where the server program resides. For example, if the server is called Qedit.Pub.Robelle, the trace files will be

```
access.qlog.robelle
error.qlog.robelle
trace.qlog.robelle
```

On UNIX hosts, the trace files are located in a directory called `log/qedit` under the directory specified in the ROBELLE environment variable. See the *Qedit for HP-UX User Manual* for details. For example, if the server is in `/opt/robelle/bin/qedit` and the ROBELLE environment variable is not set, the trace files will be

```
/usr/robelle/log/qedit/access.log
/usr/robelle/log/qedit/error.log
/usr/robelle/log/qedit/trace.log
```

The server trace facility can be enabled by checking the **Enable trace** option of the **Server Trace Options**. The facility is then activated for all connections initiated by this user.

When debugging connection problems, at a minimum, the following options should be enabled:

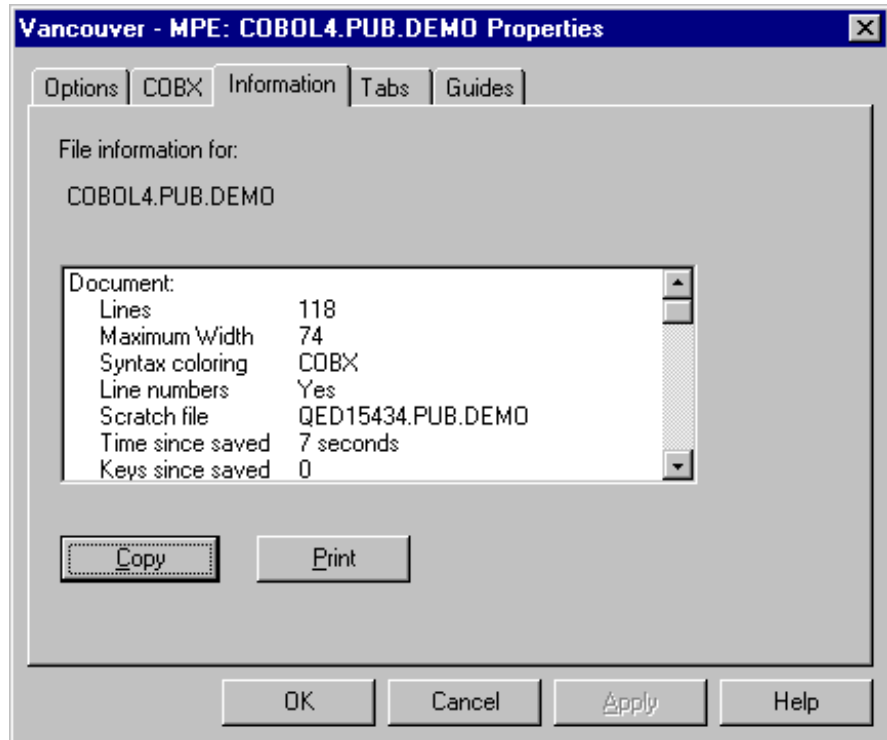
- Messages from host
- Messages to host
- Raw buffers from host
- Raw buffers to host

Changing Debug Settings

If you wish to change the debug settings temporarily, click the **Apply** button. The new settings are effective from that point but are only valid for the duration of the current Qedit for Windows instance. As soon as you quit, Qedit returns to the permanent settings. If you want the new settings to be permanent, click **Save**.

File Information Dialog Box

Qedit for Windows keeps track of certain information about the connection and the document you are currently working on. This information can be analyzed to identify potential problems. To view the information, select the **File** menu, and point to **Properties**. When the **Properties** dialog box appears, click the **Information** tab.



File Information page on the File Properties dialog box

The information in this dialog box is document-specific. The dialog box above shows information about Cobol4.Pub.Demo. You can look at the details by using the scroll bar.

If you want to capture the information, you can select **Copy** and put all these items in the Windows Clipboard.

You can print this information by clicking on **Print**, which will give you the standard Windows **Print** dialog box.

The Clipboard and the printed document will contain additional information about Qedit for Windows (edition, version number and build timestamp) as in this example:

```
Qedit 32-bit Edition Version 4.5.24 Sep 19 1997 23:03:07
```

Select **Done** to close the dialog box.

The information is divided into five (5) sections: Document, Connection, Transmission, Server Settings, and Cache Statistics. Section headings are indented from the related details for easy reading.

Document Information

The first section provides file attributes. Details for local files differ slightly from host files. Qedit displays the fully qualified file name in both cases. For local files, this means identifying the disk drive and all directories leading to the file. For native MPE host files, the file name appears in the `<file>.<group>.<account>` format. For files on HP-UX or in the POSIX name space on MPE hosts, the name is displayed in the UNIX format starting from the root directory.

Next, Qedit displays the number of lines currently in the file.

Then, Qedit displays the maximum line width allowed in the file. This corresponds to the **Record length** parameter in the **File Options** dialog box.

Syntax coloring is the name of the language assigned to this file according to the corresponding language template. If the file is not associated with a template, Qedit displays "Unknown".

For host files only, Qedit displays the name of the scratch file used.

- Local file

```
Document:
  C:\Robelle\Qedit\roman8.txt
  Lines 8
  Maximum Width 2045
```

- MPE host file

```
Document:
  TEST1.PUB.DEMO
  Lines 10
  Maximum Width 80
  Scratch file QED77264.PUB.DEMO
```

- HP-UX host file

```
Document:
  /users/demo/test1
  Lines 10
  Maximum Width 1000
  Scratch file /usr/tmp/qscr.CAAa22757
```

Connection Information

The Connection information section is only applicable to host files. Qedit displays the same pieces of information whether you are connected to an MPE or HP-UX host.

Qedit displays the **Connection** details used when you created this connection: the connection name, the name of the host computer and the login information.

Next, it displays some information about the server. It shows the version of the Qedit server program in use followed by the type and version of the operating system on the host.

Each child process created by the server program has its own process ID. This number is used to identify all transactions logged in the server log files. All server logfile entries related to a particular connection are identified by this process ID. It appears in square brackets on each line. For example, the following entries are for the server process 187:

```
Sep 22, 1997 09:31:39 [187] TRead. Length=12,KeepAlive,version=1,id=0
Sep 22, 1997 09:31:39 [187] TSent. Length=12,KeepAlive,version=1,id=0
```

Qedit then provides some statistics about the network traffic and the amount of information transferred since the beginning of the edit session. These statistics include the number of packets sent and received by the client and the total number of bytes transferred in these packets.

```
Connection:
  Name Vancouver - MPE
  Host Name DEV
  Host User demo,mgr.demo
```

```
Host Qedit   Qedit/iX (Version 4.6.01)
Host OS     MPE/iX C.55.00
Host Process ID   82
Packets Sent      3
Packets Received  4
Bytes Sent       132
Bytes Received   812
```

Transmission Information

The transmission section is really made up of two subsections: the Transmit Queue and Connection Response Time.

The packets sent to the server are submitted using an asynchronous scheme. This means Qedit for Windows writes outbound packets in a queue and, in most cases, does not wait for an answer from the server. Depending on the network load, these packets might temporarily remain on your PC. These packets are stored in memory in an area of limited size.

In the Transmit Queue, the queue length indicates how many packets are currently waiting for transmission. This number should change every time you look at the **File Information** dialog box.

The highwater mark value indicates the highest value ever attained by the queue length.

The capacity is the absolute maximum number of packets that can be queued.

```
Transmit Queue:
Length      0
Highwater Mark 64
Capacity 16384
```

The Connection Response Time section gives some statistics on the overall performance of the network. Once every minute, Qedit sends a transaction to the host to ensure that it is still processing transactions. Because these transactions require no processing on the host, just an acknowledgment, they are the ones used to produce the response time values.

The first two items indicate the amount of time Qedit for Windows had to wait before it received an acknowledgment from the server. The Last Transaction item indicates the delay for the last acknowledgment from the host. The Longest transaction value indicates the longest wait for an acknowledgment since the connection was established. The Connection Time indicates how long you have been connected to the host computer for the connection.

```
Connection Response Time:
Last Transaction  449 msec
Longest transaction  449 msec
Connection Time   1 minute, 32 seconds
```

Server Settings

Some server settings can be customized using the configuration files residing on the host. On an MPE host, there is `Qeditmgr.Pub.Sys` and `Qeditmgr.Pub.Robelle` plus any other optional configuration file enabled in the **Configure Server Settings** dialog box. On HP-UX, the system configuration file is called `/opt/robelle/qeditmgr` and the user configuration file is

.qeditmgr in the user's home directory. This section describes the settings for this particular connection.

```
Server settings:
  Tabs                Every 10 columns
  COBOL Tag           ME970924
  Show tags           Yes
  Fixed-length Cobol  No
```

Cache Statistics

Qedit for Windows uses a memory cache to store lines from host files. The goal is to have enough lines readily available on the PC so that the response time is not severely affected by slow connections.

Similar to host files, local files are not fully loaded into memory. Qedit just reads the records as they are needed.

This section can have two parts: Cache Statistics and Cache Requests. The statistics show both types of files. The Cache Requests subsection only applies to host files.

The Cache Statistics subsection shows the number of bytes that are actually data from the file (i.e., text that you can see on your document window and that can be edited).

Of course, Qedit needs to manage all this information and must keep some information on the structure of the cache. The space required to store this information is shown as Overhead.

Next, Qedit shows the number of individual lines that are represented by the Data bytes. It also shows how much of the cache space is actually used by all this.

The cache is not monolithic. It is actually made of smaller blocks, each containing a certain number of lines. Qedit displays the number of blocks currently in use. Some blocks might be locked. This happens when Qedit is waiting for a response from the host that involves lines in a block.

Because the cache is a critical component of Qedit, it has to ensure that everything in it is sound. On a regular basis, Qedit goes out and checks the integrity of the cache structure.

```
Cache Statistics:
  Data 14995 Bytes
  Overhead 3640 Bytes
  Lines 361
  Cache Full 75%
  Blocks 8
  Blocks Locked 0
  Integrity Calls 385
```

For local files, everything is done by the client program. For host files, the client has to request lines from the server. Thus, the client issues some requests and also counts the number of cache entries received in response. The last item shows the number of requests that have not been satisfied yet by the server.

```
Cache Requests:
  Issued 14
  Received 74
  Outstanding 0
```

Connection Problems

MPE Host Connections

In order to open a file on an MPE host, Qedit for Windows must go through a number of steps.

1. Qedit for Windows needs to resolve the host name or IP address you entered in the **Connection List** dialog box. If there is a Domain Name Server (DNS) configured on your PC, a host name is automatically converted to its corresponding IP address.

Most PCs come with a program called *ping*. It allows the testing of connections to other machines on the network. You can use it to make sure DNS is configured correctly and that the host name is reachable from your location. If you cannot use the host name, you can also try the exact IP address, if you know it.

2. When you start a new Qedit for Windows session, there is no connection unless one or more connections are configured with the **Login at startup** option. Upon opening the first file, Qedit for Windows must establish a link to the host. By default, it uses the Remote Process Management (RPM) feature of MPE/iX. RPM comes standard in MPE/iX 5.0 or later. If you are running earlier versions of MPE/iX, you have to purchase the NS/3000 product.

If your company uses firewalls and other network security devices, the RPM protocol might not work. If some of your connections require Qedit's firewall protocol, refer to "Firewall Protocol on MPE Connections" on page 213 or the appropriate host-based Qedit manual for complete server installation instructions.

3. Qedit sends the login information (*Jobname*, *User.Account*, *Group*) and passwords, if any. RPM validates this information and returns a confirmation message. If there is an error, RPM returns a generic error message. It does not provide any hints as to the cause.

If the login information is complete, RPM initiates a new programmatic session on the HP e3000. These sessions, however, are not identical to interactive sessions initiated on terminals. RPM sessions do not count against the system's user license, and logon UDCs are not executed.

If you are using Security/3000 from VESOFT and you want to implement the same level of security for network logons, you need to run a job called `BACKG` at all times. Please refer to Security/3000's documentation for more details. Qedit for Windows works with Security/3000 as long as you are using passwords for job names, users, accounts and/or groups. Dynamic password prompting is not supported. You cannot send messages using MPE's `Tell` command either.

FTP's implementation on MPE also uses RPM as a login mechanism. If you have difficulty connecting with Qedit for Windows, you should try to establish an ftp connection. If the same problem occurs with ftp, you should focus your attention on the HP3000 security settings. If ftp works, the problem lies between the Qedit client and server.

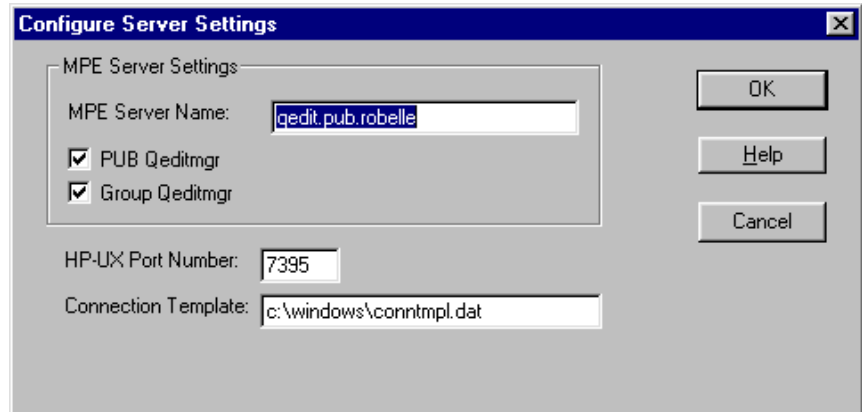
4. Qedit tries to determine the PC's IP address. The method used is the same as the standard method on UNIX. It relies on calling routines named *gethostname* and *gethostbyname* in order to translate the PC node name to a number.

The IP address is typically a fixed value for a given PC. However, if you are connecting via an ISP or some dial-in equipment, the IP address might change on every connection. Qedit will re-assess the IP address every time it tries to establish a new connection.

5. Next, Qedit runs an instance of MPE's command interpreter (CI), Ci.Pub.Sys. It passes a command to the CI so that the Qedit server program gets started.

If you are using a trial version, the MPE Server Name should be Qedit.Pubtri.Robelle.

The server program name is determined by the **Server** command in the **Options** menu. The default name is Qedit.Pub.Robelle. This is where host-based Qedit is usually installed.



Configure Server Settings dialog box

If the connection process seems to hang at the “Waiting for server” message, make sure you have configured the right **MPE Server Name** and check the host to make sure the program actually resides at that location.

The server program receives information via an Info= <string>. The -d argument is used to pass the IP address of the PC and the connecting port number. This information is passed as a hexadecimal value, which will look like this:

-d0x1A2B3C4D5E6F

The first eight characters represent the IP address where 2 characters are used for each part of the address. In this example, the IP address would be 26.43.60.77:

\$1A = 26
\$2B = 43
\$3C = 60
\$4D = 77

The port number is assigned by Winsock from the pool of available ports on the PC, and changes from session to session. The port number is represented by the last four characters of the argument. In this example, it is:

\$5E6F = 24175

Depending on options selected in the **Debug Controls** dialog box of the **Options** menu, other parameters will be passed to the CI process.

6. At this point, Qedit for Windows turns itself into a daemon and waits for confirmation from the server.

Server debugging mode can be enabled in the **Debug Controls** dialog box.

If you are using a trial version, the server name is *Qedit.Pubtri.Robelle* thus the log files will be in *Qlogtri.Robelle*.

7. On the host, the server initializes its environment. One of the first things it does is open the log files. If it cannot open them, it prints a message on the console and terminates.

These log files are used for keeping track of the server activities and, optionally, recording additional events for debugging purposes.

On MPE hosts, the log files are created in a group called Qlog which is in the same account as the server program. For example, if the **MPE Server Name** in the **Configure Server Settings** dialog box is *Qedit.Pub.Robelle*, the log files will be in the group *Qlog.Robelle*. The log files are called Error, Access and Trace.

8. The server tries to open an outbound link on the port number specified in the `-d` parameter, back to the originator's IP address.

If the outbound link cannot be established, the server writes error messages to the log files and terminates. Use `NSCONTROL STATUS` to ensure the RPM and RPML services are running on the host. Because the outbound port number is assigned on the PC and the number changes for every connection, you might run into problems if you are using a firewall at your site. You might have to reconfigure your firewall to allow the use of these ports. Unfortunately, there is no easy way to determine in advance what these port numbers will be.

9. If the link is established, the file is opened and data is transferred between the server and the client.

Firewall Protocol on MPE Connections

The firewall protocol on HP e3000 hosts borrows traits from the RPM protocol and the HP-UX server protocol. RPM is part of the MPE operating system and, as such, is mostly transparent. If you use the firewall protocol, you have to manually run the Qedit server in a special mode. It replaces part of the RPM process and takes care of managing incoming connection requests.

In order to open a file on an HP e3000 host running with the firewall protocol, Qedit for Windows must go through a number of steps.

1. You have to start the Qedit main server process on the host. For detailed installation instructions, refer to the Qedit for MPE User Manual.

Start the server program with the `-d` argument.

```
run qedit.pub.robelle;info="-d 7395"
```

Note that a space is required between the argument name, `"-d"` and the port number, `"7395"`. The number besides the argument is the port number where incoming Qedit for Windows connection requests are sent to. The main server process is listening for requests on that port.

Typically, this is done inside a job stream that runs continuously. You can find a sample stream in `Qserver.Qeditjob.Robelle` on the host. If the job is not running, new connections cannot be established.

2. The server then initializes its environment. One of the first things it does is open the log files. If it cannot open them, it prints a message on the console and terminates.

Server debugging mode can be enabled in the **Debug Controls** dialog box.

If you are using a trial version, the server name is *Qedit.Pubtri.Robelle* thus the log files will be in *Qlogtri.Robelle*

These log files are used to keep track of the server activities and, optionally, record additional events for debugging purposes.

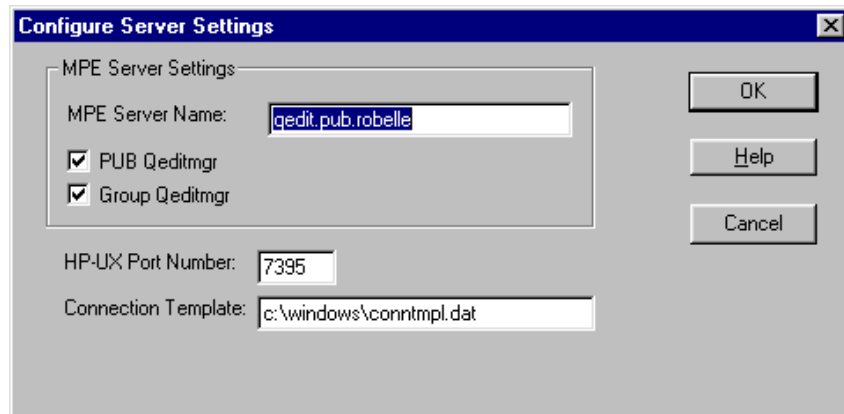
On MPE hosts, the log files are created in a group called **Qlog** which is in the same account as the server program. For example, if the **MPE Server Name** in the **Configure Server Settings** dialog box is *Qedit.Pub.Robelle*, the log files will be in the group *Qlog.Robelle*. The log files are called **Error**, **Access** and **Trace**.

3. When trying to establish a connection, Qedit for Windows needs to resolve the host name or IP address you entered in the **Connection List** dialog box. If there is a Domain Name Server (DNS) configured on your PC, a host name is automatically converted to its corresponding IP address.

Most PCs come with a program called *ping*, which can test connections to other machines on the network. You can use it to make sure the DNS is configured correctly and the host name can be reached from your location. If you cannot use the host name, you can also try the exact IP address, if you know it.

4. When you start a new Qedit for Windows session, there is no connection unless one or more connections are configured with the **Login at startup** option. Upon opening the first file, Qedit for Windows must establish a link to the host. It tries to open a connection on port number 7395. The main server program should be listening on that port, whose number has been reserved for Qedit's sole use.

You can check the configured value by using the **Server** command in the **Options** menu. Remember that the port number must match the number used by the server process on the host.



Configure Server Settings dialog box

You can use the *telnet* command with the host name and port number to make sure the main server is running. Just type

```
telnet hostname 7395
```

If you get the following message, the server is not running at all or is using a different port number:

```
Trying...
telnet: Unable to connect to remote host:
Connection refused
```

However, if you get a dialog box similar to this, and you press the carriage return key a few times, there is a very good chance the main server is active:

```
Trying...
Connected to prodmpe.robelle.com.
Escape character is '^]'. <cr>
<cr>
07B<cr>
<cr>
A<cr>
Connection closed by foreign host.
```

5. The local node name must be configured correctly on the host. The node name must appear in the **Local HP 3000 Node Name** field on the **Config** entry screen and also in the **Local Domain Name** field on the **NS Configuration** screen. You must be able to run the NMMGR network configuration program on the HP e3000 to check (and modify) these values.
6. Qedit sends the login information (*Jobname, User.Account, Group*) and passwords, if any. The client socket number is also passed in the transaction.
7. The main server gathers the connection information and invokes RPM. RPM takes over from that point.
8. RPM validates this information and returns a confirmation message. If there is an error, RPM returns a generic error message. It does not provide any hints as to the cause.

If the login information is complete, RPM initiates a new programmatic session on the HP e3000. These sessions, however, are not identical to interactive sessions initiated on terminals. These sessions do not count against the system's user license, and logon UDCs are not executed.

If you are using Security/3000 from VESOFT and you want to implement the same level of security for network logons, you need to run a job called `Backg` at all times. Please refer to Security/3000's documentation for more details. Qedit for Windows works with Security/3000 as long as you are using passwords for job names, users, accounts and/or groups. Dynamic password prompting is not supported. You cannot send messages using MPE's `Tell` command either.

FTP's implementation on MPE also uses RPM as a login mechanism. If you have difficulty connecting with Qedit for Windows, you should try to establish an ftp connection. If the same problem occurs with ftp, you should focus your attention on the HP3000 security settings. If ftp works, the problem lies between the Qedit client and server.

9. Next, RPM runs an instance of MPE's command interpreter, `Ci.Pub.Sys`. It passes a command to the CI so that the Qedit server program gets started.

The server program name is determined by the **Server** command in the **Options** menu. The default name is `Qedit.Pub.Robelle`. This is where host-based Qedit is usually installed.

You have to make sure the Qedit server name in the **Configure Server Settings** dialog box is the same as the one used to start the main server. This ensures you are running the same version of the server.

10. If the connection is established, the file is opened and data is transferred between the server and the client.

If you are using a trial version, the MPE Server Name should be `Qedit.Pubtri.Robelle`.

HP-UX Host Connections

In order to open a file on an HP 9000 host, Qedit for Windows must go through a number of steps.

1. You have to start the Qedit for HP-UX server on the host. For detailed instructions, refer to the Qedit for HP-UX User Manual.
 - Log in as root.
 - If you have installed Qedit in a directory other than the default /usr/robelle, you have to set the ROBELLE environment variable with the actual install directory.

```
ROBELLE=/users/apps
```

You also need to export the variable.

```
export ROBELLE
```

- Start the server program with the -d argument.

```
/opt/robelle/bin/qedit -d
```

2. The server then initializes its environment. One of the first things it does is open the log files. If it cannot open them, it prints a message on the console and terminates.

These log files are used to keep track of the server activities and, optionally, record additional events for debugging purposes.

By default on HP-UX hosts, the log files are created in a directory called log/qedit, which is under the main Robelle directory. For example, if the server program is in /opt/robelle/bin, the log files will be in the /opt/robelle/log/qedit directory. The log files themselves are called error.log, access.log and trace.log.

If you have set the ROBELLE environment variable, the log files will be created in the specified directory. You have to ensure the new location has the same structure as the original /opt/robelle.

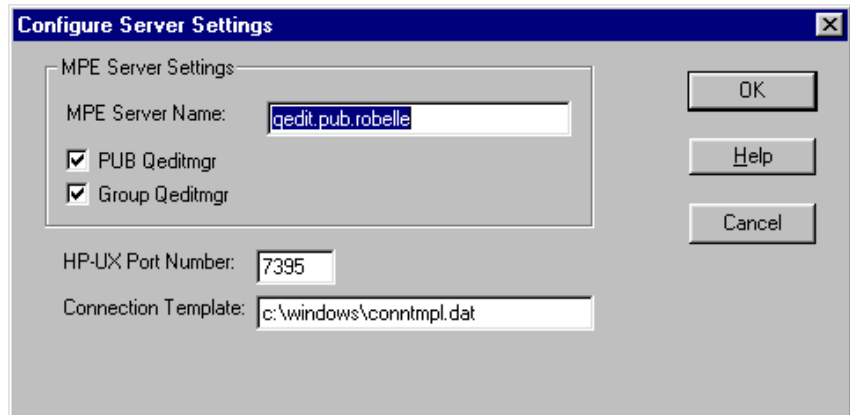
3. Qedit for Windows needs to resolve the host name or IP address you entered in the **Connection List** dialog box. If there is a Domain Name Server (DNS) configured on your PC, a host name is automatically converted to its corresponding IP address.

Most PCs come with a program called *ping*, which can test connections to other machines on the network. You can use it to make sure the DNS is configured correctly and the host name can be reached from your location. If you cannot use the host name, you can also try the exact IP address, if you know it.

4. When you start a new Qedit for Windows session, there is no connection unless one or more connections are configured with the **Login at startup** option. Upon opening the first file, Qedit for Windows must establish a link to the host. It tries to open a connection on port number 7395. The server program should be listening on that port, whose number has been reserved for Qedit's sole use.

You can check the configured value by using the **Server** command in the **Options** menu. Remember that the port number must match the number used by the server process on the host.

Server debugging mode can be enabled in the Debug Controls dialog box.



Configure Server Settings dialog box

You can use the *telnet* command with the host name and port number to make sure the main server is running. Just type

```
telnet hostname 7395
```

If you get the following message, the server is not running at all or is using a different port number:

```
Trying...
telnet: Unable to connect to remote host:
Connection refused
```

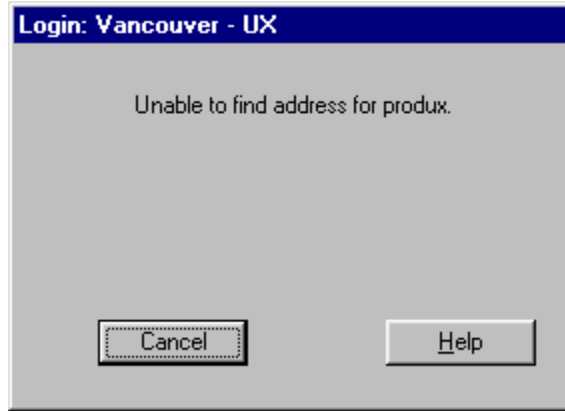
However, if you get a dialog box similar to this, and you press the carriage return key a few times, there is a very good chance the main server is active:

```
Trying...
Connected to produx.robelle.com.
Escape character is '^]'. <cr>
<cr>
07B<cr>
<cr>
A<cr>
Connection closed by foreign host.
```

5. The main Qedit for HP-UX server then uses *fork* to create another Qedit for HP-UX child process.
6. Once a connection is made between the Qedit client and the Qedit server program, the client sends a login transaction to the host. This information is validated against the host */etc/passwd* file. If there is an error, a message is sent to the client and written to the log files. The child process then terminates.
7. If the login information is correct, the server process changes its user id and group id to the ones that correspond to the user name defined in the connection.
8. Using a *cd* command, the server then goes to the home directory for that user as defined in the */etc/passwd* file.
9. The server picks up a lot of information from the */etc/passwd* file, but it does not run the default shell and does not set any of your default environment variables such as *PATH*, *TERM*, etc.
10. If the link is established, the file is opened and data is transferred between the server and the client.

Connection Progress Messages

When the user instructs Qedit to log on to a host by opening a file on an unused connection or connecting with the **Login at startup** option enabled, Qedit for Windows begins a multistep operation to complete the request. Each of the steps is accompanied by a progress message in the **Login** dialog box.



Login dialog box

The progress messages are designed to be friendly to the user, but as a result are not as explicit to a support engineer. In addition, if any step fails, Qedit for Windows issues a user-friendly message rather than a technical one. Here is the meaning of the user-friendly messages, translated into technical terms:

- Finding address for *<hostname>*.
Qedit for Windows has issued a DNS request to resolve the host name. This message will not appear if the user has specified an IP address.
- Connecting to *<hostname>*.
Qedit for Windows has issued a TCP connection request to the host. For an MPE login, Qedit for Windows requests a connection to the RPM server; for a UNIX login, Qedit for Windows connects directly to the Qedit for HP-UX server.
- Logging in...
(UNIX only.) The connection is up and Qedit for Windows is sending the login transaction. This message might be flashed too briefly to see.
- Awaiting welcome...
(UNIX only.) Qedit for Windows has sent the login transaction and is waiting for the server's response. When the response arrives, the login transaction is complete (though the login attempt might have been rejected).
- Logging in and starting server...
(MPE only.) The connection to the RPM server is up and Qedit for Windows has sent the RPM request. The RPM request contains both the login parameters and the name of the program to run, hence the message covering both. Qedit for Windows is now waiting for a response from the RPM server.
- Logged in. Waiting for server response...

(MPE only.) The RPM server has responded that the login and process start were successful. Qedit for Windows is now waiting for the new server process to connect to it. On MPE, the server initiates a connection to the client, rather than the other way around.

- **Login Failure Messages**

When the login process fails due to a user error (e.g., an incorrect password), the message should be self-explanatory. However, login failures due to network problems cause Qedit to generate a user-friendly message. The message is of the form “Unable to... because...” where the first ellipsis is replaced by the action that failed and the second is replaced by a translation of the error code that Qedit received. Because these messages indicate a low-level technical problem, they are difficult to document. You should get assistance from our technical support staff to get these resolved. Here are some examples of login failure messages:

- Unable to obtain a socket because...
- Unable to set up receive connection because...
- Unable to set up the connection because...
- Connection failed because...
- Unable to obtain resources for an MPE Login because...
- Unable to start MPE server because a problem occurred.
- Unable to complete login because...
- Unable to start communication because...
- Login failed because...

Recovering a File

Recovering Temporary Host Files

When working on host files, Qedit creates a temporary holding area called a scratch file. These scratch files, which are Qedit workfiles, write all the changes you make to disc almost immediately. These files are temporary in the sense that they will be purged when the connection is closed normally. If the connection closes abnormally, these files still exist on the host computer. These combined features provide a recovery mechanism if a connection failure occurs.

Recovering a File on MPE Hosts

On MPE hosts, scratch files are called QED##### where the number signs represent a 5-digit random number (e.g., QED24070). They are created in the logon group for the connection used at the time. To recover your latest changes,

1. Go to the **Directory** window for the connection you were using.
2. Look for Qedit scratch files (QED#####).
3. If there is more than one scratch file, look at the **Last Modified** column and find the one that is more likely to be yours (today’s date and time). There will be more than one if there are other users connecting with the same logon ID (group and account).

NOTE: The **Directory** window you are looking at is also stored in a Qedit scratch file. Make sure you select the right one. If you select the directory scratch file, you will likely get an open error.

4. Open the file. Qedit automatically performs an internal recovery process. You should then have the text you were working on.

If the Qedit client aborted, it is possible that you will not be able to open the file immediately. That's because the server uses an internal timeout of 5 minutes before it assumes there is a problem with the connection. So, for the timeout duration, scratch files and Qedit workfiles remain open and cannot be accessed.

5. If the text is yours and in good shape, perform a **Save as** command. Because this is a Qedit workfile, the server simply renames it from QED##### to whatever name you assign. The new file is still a Qedit workfile. If you want to save it as a regular file, you should open a copy of the original workfile and then do a **Save as** command.
6. Once you are sure that you have recovered everything, you can delete the scratch file.

Recovering a File on HP-UX Hosts

On HP-UX hosts, scratch files are created in your environment temporary directory. The temporary directory is usually defined with the TMPDIR environment variable. If you do not know what it is, you might have to ask your system administrator or connect using a terminal and type:

```
$ echo $TMPDIR  
/usr/tmp
```

In this example, the environment temporary directory is /usr/tmp. The scratch files are called qscr.????????? where the question marks can be mixed case letters and numbers (e.g., qscr.CAAa24070).

1. Go to the **Directory** dialog box for the connection you were using.
2. Go to your environment temporary directory. You can navigate by double-clicking the appropriate entries in the directory listing or you can go directly to it by entering the directory name (e.g., /usr/tmp/) in the file name text box and selecting **Open**.
3. Look for Qedit scratch files (qscr.?????????).
4. On most systems, all the users share the same environment temporary directory so there will likely be more than one scratch file. Look at the **Last Modified** column and find the one that is more likely to be yours (today's date and time).

NOTE: The **Directory** window you are looking at is also stored in a Qedit scratch file. Make sure you select the right one. If you select the directory scratch file, you will not get an error but the contents will be almost unreadable.

5. Open the file. Qedit will automatically perform an internal recovery process. You should then have the text you were working on.
6. If the text is yours and in good shape, perform a **Save as** command.
7. Once you are sure you have recovered everything, you can delete the scratch file.

Local Files Operation

When you open a local file for editing, Qedit reads the entire file. This is done for two reasons:

1. To count the number of lines
2. To create an internal table of line pointers into the original file. This table requires 16 bytes per line in the file.

Qedit keeps track of both the number of bytes in the file and the number of lines in the file. Due to internal constraints, the maximum size for a local file is:

1. Four billion characters or about 4 GB (2^{32}).
2. With up to 2 billion lines ($2^{31} - 1$).

The file is read in 20,480-byte chunks. The number of lines is used to move the scroll box to the appropriate location in relation the file size. If the scroll box is about $\frac{3}{4}$ of the way down the scroll bar, the displayed lines are really about $\frac{3}{4}$ of the way down the file. Other editors use the number of characters to position the scroll box, which might not be correct if you have very short or very long lines.

Changes that you make to the file are retained in memory. A global change that affected every line in a 2-GB file would not fit in memory. In this case, Qedit shows a warning to let you terminate, possibly without saving your changes. This is a function of the virtual memory emergency code in Qedit and not anything to do with local editing per se.

When you save your changes, the following takes place:

1. Qedit creates a temporary file on the same drive and in the same directory as the original file with the name QSnnnnnn.tmp, where *nnnnnn* is replaced with a random number.
2. The changes made are merged with the original file to create a new copy. This is similar to performing a **Text** and **Keep** command in host-based Qedit. Even if a single line was changed, the entire original file must be copied.
3. The original file is purged.
4. The temporary file is renamed as the original file.

Time Zones

You pull up your home group for the development HP e3000 and you notice that the modify date for a file you know you changed ten minutes ago is way off—4 hours later, in fact. What has gone wrong?

Greenwich Mean Time (GMT/UTC) is the answer!

Client/server applications, including Qedit for Windows, use GMT to record times. This is the time in Greenwich, England, ignoring Daylight Saving Time, and is considered zero (0). New York is normally at GMT-5, Los Angeles is at GMT-8. Paris and Berlin are at GMT+1 and Athens is at GMT+2.

For a map showing the whole thing, go to this Web site:

<http://www.crasys.anu.edu.au/reference/world.timezones.html>

If all network timestamps are in GMT, how does your computer tell you the actual local time when something happened?

Does each computer have to know what time zone it is in?

This is done by setting a TZ variable on the host computer. UNIX does this automatically when it boots up (although it can still be wrong!), while MPE must do it manually for each user.

Consider this recent case and try to figure out what went wrong. A file called Quality in a home group on an MPE system in Western Canada looked like this:

```
:listf quality,3
...
NUM LABELS: 2 CREATED : FRI, AUG 6, 1993, 9:26 AM
MAX LABELS: 2 MODIFIED: SAT, SEP 20, 1997, 8:08 AM
DISC DEV #: 1 ACCESSED: SAT, SEP 20, 1997, 8:08 AM
```

But in Anguilla in the Caribbean, which is 3 hours ahead of Vancouver at this time of year, the modification time in a Qedit directory box should have been 11:08 A.M. When the modify time was checked by running Qedit and opening the group listing, it showed 4:08 P.M. on the same day. The difference was *four hours*.

The value of the TZ variable was checked on the Vancouver machine.

```
:showvar TZ
TZ = PST8
```

PST means Pacific Standard Time (GMT-8), which can be checked in the main pages on the UNIX system.

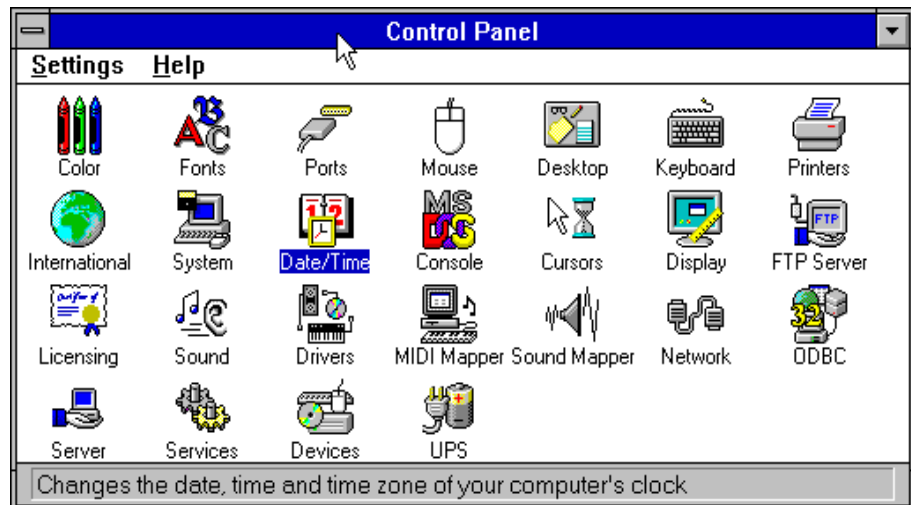
When the time on the Vancouver system was checked, it looked correct:

```
:showtime
THU, SEP 25, 1997, 10:19 AM
```

When the clock on the Anguilla PC (Windows NT 3.51) was checked, it also showed the correct local time. Although Windows 95 doesn't have a clock program, it does show the time at the right edge of the taskbar.

What else is there to check?

The time zone configuration on the Anguilla PC was checked. In Windows NT 3.51, the **Date/Time** was checked on the **Control Panel** program group.



Windows NT 3.51 Control Panel program group

The **Date/Time** showed that the Anguilla PC was set to Greenwich England, or some place in that time zone!



Windows NT 3.51 Date/Time control panel settings

The time zone had to be adjusted to Atlantic Time (Canada). The automatic adjustment had to be disabled for Daylight Saving Time, because Anguilla never goes on Daylight Saving Time.

To check the file's modified time, Qedit was started again. The modify time now showed 12:08 P.M. This was closer, but it was still one hour off. It should have been 11:08 A.M. Perhaps the problem was that the TZ variable on the Vancouver host was set to PST8 instead of PDT7. How can the TZ variable be checked?

How can the variable be set for the MPE Qedit server?

Put this command in your Qeditmgr.Pub.Robelle file:

```
setvar TZ "PST8"
```

UNIX options specifying Daylight Saving Time

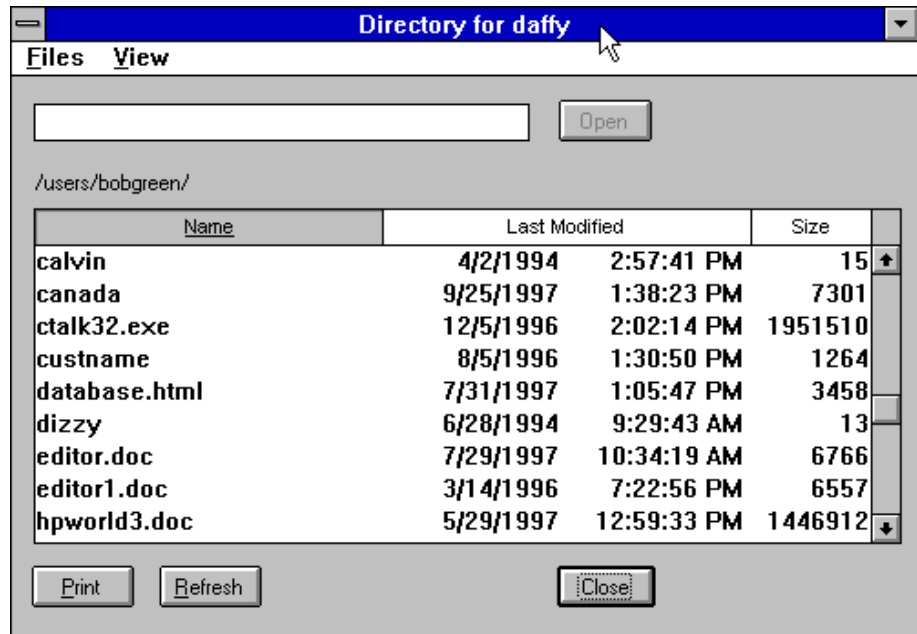
To check the Qedit modification times on the UNIX server in Vancouver, *telnet* was used to access the West Coast UNIX system and *touch* was used to update the timestamp of the file. Then the file was listed in the UNIX shell:

```
/users/ralph $ ll canada
-rw-rw-rw- 1 ralph users 7301 Sep 25 10:38 canada
```

A 10:38 A.M modification time in Vancouver is 1:38 P.M. in Anguilla. Just for thoroughness, the time zone variable and the current time on the UNIX system were also checked:

```
/users/ralph $ echo $TZ
PST8PDT
/users/bobgreen $ date
Thu Sep 25 10:45:42 PDT 1997
```

Now the time was correct and showed that it was PDT. The TZ variable was a little different from the MPE system. It seemed to be saying that the time zone is GMT-8, but the system also recognizes Pacific Daylight Time.

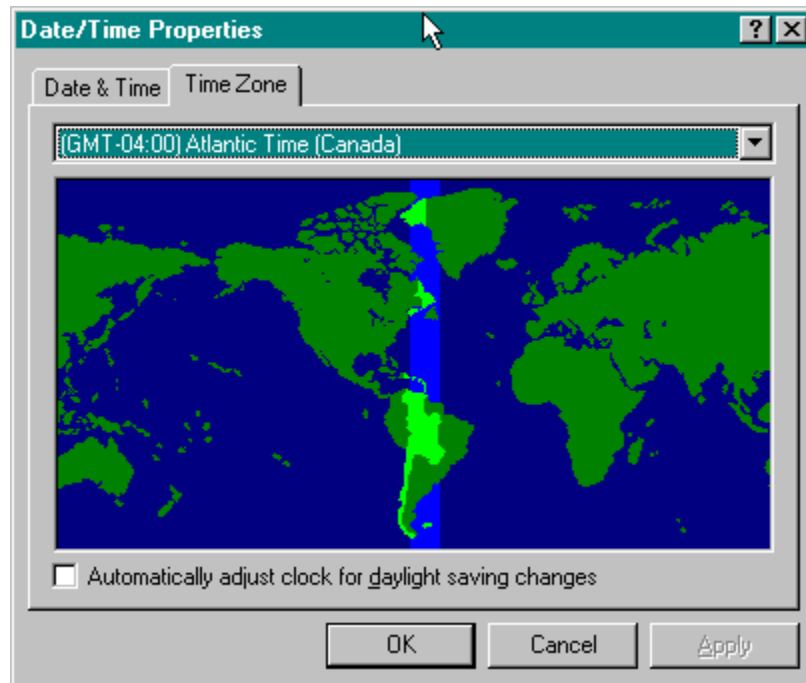


Directory dialog box for the HP-UX host in Vancouver, British Columbia, Canada

As you can see, the **Last Modified** time for the canada file was 1:38 P.M., which is what was expected.

Checking the Time Zone on Windows 95

To set the time zone on Windows 95, select the **Start** button, point to **Settings**, and click **Control Panel**. When the **Control Panel** program group appears, click the **Date/Time** icon. Go to the **Time Zone** tab.



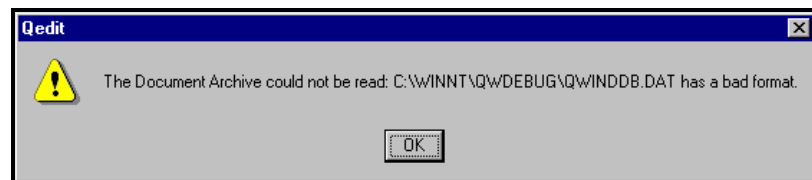
Windows 95 Date/Time control panel

The Windows 95 PC above seems to have the correct time zone and is *not* adjusting for Daylight Saving Time. In the USA and Canada, anyone who lives outside of Central Standard Time will need to enable this option.

Document Database Recovery

Qedit for Windows stores information about each file in a document database called QWINDDB.DAT. Under some circumstances, the document database may become corrupt. If this ever happens, Qedit for Windows will not be able to start. Starting with version 5.4.12, Qedit asks the user for a course of action.

One of the first things Qedit does when it starts up is open the document database and load its contents. If the procedure succeeds, the current document database is copied to QWINDDB.BKP. An error message like the one below is displayed if the procedure detects a problem.



Document database problem error

You should write down the exact error message. Click **OK** to proceed with the next step. Qedit then prompts for an action to take.



Document database recovery options

*When the **Delete** option is selected, all file information previously stored is lost.*

Whichever option you chose, Qedit makes a copy of the bad document database to QWINDDB.DAT.BAD. The **Restore** option is enabled only if there's a backup copy available. Select **Delete** to get rid of the current document database and start with a new one.

Select **Restore** to return to the last known valid database. Qedit copies qwinddb.bkp back to qwinddb.dat and tries to load it. If this fails again, an error message is displayed and the document database is erased.

In any case, you should report the problem to Robelle technical support with the exact error message and, if at all possible, send the bad document database along for further analysis.

Error Messages



Fatal error

From time to time, Qedit needs to let you know that something is not working quite right. As much as possible, you will only see a message box with a short message describing the problem as accurately as possible. There are two types of messages: warning and fatal. Each type is identified by an icon appearing in the message box.

Qedit always checks internal structures to make sure nothing is wrong. If it ever finds something unexpected, Qedit displays an error message and aborts. This might sound drastic but it is the only way to prevent further damage.



Informative or warning message

From time to time, Qedit also displays messages to show informative text or less serious errors. Typically, these messages inform the user about the success or failure of an edit operation. These messages are just warnings because there is no threat to the physical integrity of the file.

Here are some of the messages that you might encounter:

- Cannot continue editing on connection *<connection name>* because the host reset the connection. The host process identifier for this connection was *<pid>*.
Qedit detected a problem with the connection to the host. The server is not acknowledging transactions anymore. The Process Identification number (PID) can be used to search through the server log files when trying to determine the cause of the problem.
- Error deleting a local file—permission denied.
You are trying to save a local file that has been opened as **Read Only** by another application or in another window in Qedit.
- Login: Invalid user or password.
The user name or password entered on an HP-UX connection is incorrect.
- Login: Unable to change directory.
Qedit could not switch to the directory you specified in the Initial directory text box in the Connection List dialog box.
- System command failed. (Server error 37)
The server has tried to execute an operating system command but the command failed. You might receive this message if you have tried to open an invalid directory name from the **Directory** window.
- The host file has been modified by another user. (Server Error 46)
This message appears when you try to do a **Save** on a host file. It means the file you are editing has been modified since you opened it or the last time you saved it.

You should try to find who has modified the file so you do not lose someone else's changes. You can also try to find what has changed. You can save your copy of the file with a different name and then compare it with the copy on the host. Host-based Qedit comes with a program called Compare which lists differences between 2 files. Look at the output and determine if the differences are valid.

If you want to force Qedit to save the file anyway, you can disable the **Compare timestamps before overwriting on server** option in the **Preferences** dialog box or use the **Save as** command and enter the original file name. Remember that changing the **Preferences** option will be applied to all your files after that.

- The resulting line would be too long.

The edit operation you are trying to perform would cause the current line to exceed the defined maximum length of the file. This message often appears if you try to combine two lines by deleting the end-of-line character or paste some text from the Clipboard.

- The server file is full. (Server error 38)

Unlike local files, files on MPE hosts can have a preset limit thus you cannot add new text indefinitely. One way to get around this error is to

1. Close the file.
2. Open the same file using **Open a copy**.
3. Open the **File** menu and select **Save as**.
4. Type the original file name in the **File name** text box.
5. Select **Save as Qedit workfile**.
6. Click **Save**.

- WinHelp cannot find the Help file. A newer version of help is needed.

Make sure you have the correct Help files for the version of Qedit you are using. Typically, you get this error if you have the 16-bit Help file while using the 32-bit Qedit program or vice-versa.

- Unable to complete login: the host refused the connection.

The Qedit for HP-UX server program is not running or you have configured a different port number.

- Unable to find address for *<hostname>*.

Qedit for Windows was unable to get the IP address for the specified host.

- Unable to open *<file name>* on *<connection>*.

The server is not able to open the file on the connection. In some cases, Qedit displays additional information on the cause of the problem. For example, NONEXISTENT PERMANENT FILE (FSERR 52) can appear. If you are trying to open a Qedit workfile that is already being accessed, you might not get an explanation. Make sure the file exists and is accessible from a terminal using host-based Qedit.

- Unable to start MPE server because the supplied login was not valid.

This message is displayed when RPM cannot initiate a new session. RPM returns a generic error code. Qedit cannot establish the exact cause of the

problem. Anything that can prevent a session from starting will return this error. Some possible causes include:

- the login information (user, group, account) in the connection is incorrect
- one of the passwords in the connection is invalid
- the session limit on the host is too low
- the string entered in the **Hello** text box does not have the right syntax
- An RPM error on the host during login (server error 75)

This message is displayed when RPM cannot initiate a new session on an MPE connection using the **Firewall** protocol. Check the **Hello** string for that connection and make sure all passwords are correctly entered.

Appendix A - Earlier Highlights

Overview

Here are some of the changes and enhancements made in earlier versions of Qedit for Windows.

There are no pre-release versions at this point.

Appendix B - File Types

File Codes and Mnemonics

The **Directory** dialog box can display a file type when working with an MPE host. The file types are short mnemonics that are somewhat easier to understand than numbers. The following table shows the MPE file codes, the mnemonic associated with each code and a short description of the type. The information in the table is based on a list of known codes graciously provided by Stan Sieler of Allegro Consultants Inc.

MPE already uses mnemonics to identify file types. These standard mnemonics appear in uppercase in the table.

MPE allows user-defined file codes. Different mnemonics have been assigned to these codes. These mnemonics appear with mixed-case or all lowercase characters.

File code	Mnemonic	Usage
-21074	vt	RoadRunner virtual tape file (priv)
-16976	vt	BackPack virtual tape file
-7652	tdir	ORBiT Backup+ tape directory saved to disk
-1234	emlg	MPE V Memory log (MEMLOG.PUB.SYS)
-999	oddf	ORBiT Backup+ disk-to-disk file
-995	oddb	ORBiT Backup+ disk-to-disk backup
-993	oddd	ORBiT Backup+ disk-to-disk directory
-893	oracl	Oracle file?
-601	tmles	TEACHME/3000
-516	aif	AIFKUF.PUB.SYS
-497	lflf	ALLBASE (DBCORE) Logfile
-493	ptpt	DBCORE Chunkset
-491	dbe	DBCORE database environment
-451	daalf	Created by NETUTIL ??
-450	dausf	DAUSF is the NETUsers config info
-431	atc	ATC file links ALLBASE DBE to TurboIMAGE root
-430	tc	TC file links TurboIMAGE root to ALLBASE DBE

-421	chpcf	DBCHANGE control file
-420	irecv	DBRECOV restart file
-412	iidx	TurboIMAGE .idx file
-411	itpid	TurboIMAGE TPU "data" file
-410	itpir	TurboIMAGE TPI "root" file
-409	icdf	TurboIMAGE Jumbo chunk data file
-408	iccf	TurboIMAGE Jumbo chunk control file
-407	igtx	TurboIMAGE GTX file
-406	idrb0	TurboIMAGE DRB'00 file
-405	iglob	TurboIMAGE control block file
-404	ilr	TurboIMAGE intrinsic level recovery
-403	idump	TurboIMAGE i-file and j-file
-402	irdba	TurboIMAGE remote database access
-401	idset	TurboIMAGE dataset
-400	iroot	TurboIMAGE root
-300	hpd	HPDESK: REPLY
-296	sg01	FYIMAIL, ONESTOP
-103	hbbs	HPDESK: Bulletin Board
-102	hpd	HPDESK: PACKAGE
-101	hpdf	HPDESK: FOLDER
-100	hpd	HPDESK: MESSAGE
-73	hbul	HPDESK: BULLETIN AREA
-60	hlist	HPDESK: LIST AREA
-50	hwork	HPDESK: WORK AREA
-40	hfcab	HPDESK: FILING CABINET
-30	hpend	HPDESK: PENDING TRAY
-20	hou	HPDESK: OUT TRAY
-12	dpcf	Unison: DiscMaster compressed file
-10	hint	HPDESK: IN TRAY
8	adasc	Adager Schema file
72	uhaul	UHAUL utility
99	hroli	HPDESK: Route List
111	Qedit	Qedit workfile
171	ves01	VESOFT
172	ves02	VESOFT
173	ves03	VESOFT
176	ves04	VESOFT
296	sg02	FYIMAIL
341	lzcsl	Interex CSL's LS program
430	dx430	DataExpress: reserved
431	dx431	DataExpress: reserved
432	dx432	DataExpress: reserved

433	dx433	DataExpress: reserved
434	dxpc	DataExpress: Procedure catalog
435	dxmc	DataExpress: message catalog
436	dxfc	DataExpress: file definition
437	dxct	DataExpress: catalog transfer
438	dxdc	DataExpress: dataview catalog
439	dxup	DataExpress: user profile catalog
440	dxsub	DataExpress: dataexpress subfile
441	dx441	DataExpress: reserved
442	dx442	DataExpress: reserved
443	dx443	DataExpress: reserved
444	dx444	DataExpress: reserved
445	dx445	DataExpress: reserved
446	dx446	DataExpress: reserved
447	dx447	DataExpress: reserved
448	dx448	DataExpress: reserved
449	dx449	DataExpress: reserved
634	hibsv	HiBack save files
640	c640	Cognos: QDD compiled dictionary file
641	c641	Cognos: QUICK compiled screen file
642	c642	Cognos: QUIZ compiled report file
643	c643	Cognos: QTP compiled run file
645	c645	Cognos: 3.02 or 4.00 QKGO execution-time parameters file
647	c647	Cognos: portable subfile minidictionary
648	c648	Cognos: portable subfile datafile
649	c649	Cognos: interim minidictionary created by QUIZ or QTP
650	c650	Cognos: QKGO execution-time parameters file
651	c651	Cognos: temporary sort file for QUIZ or QTP
652	c652	Cognos: compiled minidictionary file
654	c654	Cognos: QUICK debugger symbol file
655	c655	Cognos: compiled PDL dictionary file
665	lzw	Telamon: LZW program
667	clib	Contributed library compacted file
700	bt700	Boeing: mailbox
701	bt701	Boeing: jobqutil data
702	bt702	Boeing: maildata (forward facility of mail)
703	bt703	Boeing: letter "SAVE"
704	bt704	Boeing: post text
707	quark	Quest archiver squished file
709	bt709	Boeing: mailbox/mail

711	squi1	Boeing: Squished
712	bt712	Boeing: squish scratch
719	bt719	Boeing: squisher
724	bt724	Boeing: list program output
729	bt729	Boeing: listd output
730	bt730	Boeing: errors output
747	bt747	Boeing: scopy
757	squi2	Boeing: new squished file
777	pack	SolutionSoft: Compression Storage Manager
780	bt780	Boeing: folder data
781	bt781	Boeing: folder output
790	bt790	Boeing: document output (doc)
801	bt801	Boeing: director parameter
802	bt802	Boeing: director log
803	bt803	Boeing: report date file
804	bt804	Boeing: idlejob command
810	bt810	Boeing: remote punch input spool
811	bt811	Boeing: remote punch output spool
812	bt812	Boeing: remote punch status
813	bt813	Boeing: remote punch orders
814	bt814	Boeing: reserved for remote punch
815	bt815	Boeing: reserved for remote punch
816	bt816	Boeing: reserved for remote punch
817	bt817	Boeing: reserved for remote punch
820	bt820	Boeing: hogs user data summary (keep)
821	bt821	Boeing: hogs output
822	bt822	Boeing: log file directory
823	bt823	Boeing: reserved for log file routines
824	bt824	Boeing: reserved for log file routines
825	bt825	Boeing: conslog output
826	bt826	Boeing: reserved for log file routines
827	bt827	Boeing: reserved for log file routines
828	bt828	Boeing: reserved for log file routines
829	bt829	Boeing: reserved for log file routines
830	bt830	Boeing: bouncer data
831	bt831	Boeing: bouncer userlist
832	bt832	Boeing: bouncer data
839	bt839	Boeing: bouncer outfile
840	bt840	Boeing: dumpdst dump
841	bt841	Boeing: lostdisc output
850	bt850	Boeing: operator boot strap info (bootdata)
860	bt860	Boeing: report program config

861	bt861	Boeing: report program binary data
869	bt869	Boeing: report program output
870	bt870	Boeing: reserved for welcome config
875	bt875	Boeing: spoof output
876	bt876	Boeing: udcutil output
879	bt879	Boeing: welcome output
880	bt880	Boeing: jobnote
890	bt890	Boeing: backup configuration
899	bt899	Boeing: backup output
900	wqmac	WRQ Desk Direct: MACBINARY FILE
901	bt901	Boeing: cpu usage daily data
902	bt902	Boeing: cpu usage total
911	bt911	Boeing: response time daily data
912	bt912	Boeing: response time total
921	bt921	Boeing: numjob data
922	bt922	Boeing: numjob merge
947	cooky	Robelle: Cookie data file
969	i2sf	Unison: INSIGHT II Specification files
990	bt990	Boeing: system performance data
991	bt991	Boeing: system performance merge
993	wqwp	WRQ Desk Direct: WORDPERFECT
994	wqasc	WRQ Desk Direct: ASCII DATA
995	wqdat	WRQ Desk Direct: DATA
996	wqpar	WRQ Desk Direct: PARADOX TABLE
997	wqlt1	WRQ Desk Direct: LOTUS RELEASE 1
998	wqlt2	WRQ Desk Direct: LOTUS RELEASE 2
999	bt999	Boeing: porpoise output
1012	ocsc	OCS crunched file
1020	h1020	HPDESK: SLATE document
1024	USL	User subprogram library
1025	BASD	Basic data
1026	BASP	Basic program
1027	BASFP	Basic fast program
1028	RL	relocatable library
1029	PROG	Compatibility mode program
1030	NMPRG	NM program
1031	SL	Segmented library
1032	NMXL	NM XL
1033	NMRL	Native Mode Relocatable Lib
1035	VFORM	VIEW form
1036	VFAST	VIEW fast forms
1037	VREF	VIEW reformat

1040	XLSAV	Cross loader ascii (save)
1041	XLBIN	Cross loader relocated binary
1042	XLDSP	Cross loader ascii (display)
1043	uapps	USER_APPSET_FILECODE
1050	EDITQ	Edit quick
1051	EDTCQ	Edit keepq (cobol)
1052	EDTCT	Edit text (cobol)
1054	TDPDT	TDP diary
1055	TDPQM	TDP proof marked qmarked
1056	TDPP	TDP proof marked non-cobol
1057	TDPCP	TDP proof marked cobol
1058	TDPQ	TDP workfile
1059	TDPXQ	TDP workfile (cobol)
1060	RJEPN	RJE punch
1070	QPROC	QUERY procedure
1071	qwrk1	QUERY work file.
1072	qwrk2	QUERY work file.
1080	KSAMK	KSAM key file
1081	slmap	deleted in b.1.2 mit
1082	syml	deleted in b.1.2 mit
1083	GRAPH	Graph specification
1084	SD	Self describing file
1090	LOG	User logging log
1091	FTAM1	File Transfer Access and Management 1 (ISO standard 8571)
1092	FTAM2	File Transfer Access and Management 2 (ISO standard 8571)
1093	FTAM3	File Transfer Access and Management 3 (ISO standard 8571)
1100	WDOC	HPWORD document
1101	WDICT	HPWORD hyphenation dictionary
1102	WCONF	HPWORD configuration
1103	W2601	HP2601 environment
1110	PCELL	IFS/3000 character cell
1111	PFORM	IFS/3000 form
1112	PENV	IFS/3000 environment
1113	PCCMP	IFS/3000 compiled char cell
1114	RASTR	Graphics image in raster format
1130	OPTLF	OPT log
1131	TEPES	TEPE script
1132	TEPEL	TEPE log
1133	SAMPL	APS log
1139	MPEDL	MPEDCP/DRP log

1140	TSR	HPTOOLSET root
1141	TSD	HPTOOLSET data
1145	DRAW	HPDRAW slide drawing
1146	FIG	HPDRAW figure
1147	FONT	reserved
1148	COLOR	reserved
1149	D48	reserved
1150	h1150	HPDESK: HPSLATE DOCUMENT
1151	h1151	HPDESK: HPSLATE DOCUMENT
1152	SLATE	HPSLATE compressed
1153	SLATW	HPSLATE expanded
1156	DSTOR	RAPID: dictdbu store
1157	TCODE	RAPID: transact code
1158	RCODE	RAPID: report code
1159	ICODE	RAPID: inform code
1166	MDIST	HPDESK: distribution list
1167	MTEXT	HPDESK: text
1168	MARPA	ARPA message
1169	MARPD	ARPA distribution list
1170	MCMND	HPDESK: abbreviated commands
1171	MFRTM	reserved
1172	mrsvd	reserved
1173	MEFT	reserved
1174	MCRPT	reserved
1175	MSERL	reserved
1176	VCSF	reserved
1177	TTYPE	Term type file
1178	TVFC	Term vertical format control file
1192	NCONF	Network configuration
1193	NTRAC	Network trace
1194	NLOG	Network log
1195	MIDAS	reserved
1211	NDIR	was ANODE
1212	INODE	reserved
1213	INVRT	reserved
1214	EXCEP	reserved
1215	TAXON	reserved
1216	QUERF	reserved
1217	DOCDR	reserved
1226	VC	VISICALC
1227	DIF	Data interchange format
1228	LANGD	Language definition

1229	CHARD	Character set definition
1230	MGCAT	Formatted application message catalog
1235	atlas	reserved
1236	BMAP	reserved
1242	BDATA	Basic data
1243	BFORM	Basic field order form / vplus
1244	BSAVE	Basic saved program
1245	BCNFG	Configuration for default basic program
1246	BKEY	Function Key Definition
1247	BSVXL	Business BASIC/XL Program
1248	BDTXL	Business BASIC/XL XL Data
1249	BBNCM	Business BASIC/V Binary
1257	pfjob	con_fcode
1258	PFSTA	Pathflow static
1259	PFDYN	Pathflow dynamic
1265	h1265	HPDESK: MEMOMAKER DOCUMENT
1266	h1266	HPDESK: GRAPHICS/100 PICTURE
1267	h1267	HPDESK: GRAPHICS/100 CHART
1268	h1268	HPDESK: GRAPHICS/100 DATA
1269	h1269	HPDESK: PC FILE
1270	RFDCA	Revisable form dca document
1271	FFDCA	Final form dca document
1272	DIU	Document interchange unit
1273	PDOC	HPWORD/150 document
1274	h1274	HPDESK: ADVANCEWRITE DOC
1275	DFI	DISOSS Filing Information
1276	SRI	DISOSS Search Result Information
1277	h1277	HPDESK: VOICE MESSAGE FILE
1401	CWPTX	reserved
1421	MAP	HPMAP/3000 map specification
1422	GAL	reserved
1425	TTX	reserved
1428	RDIC	HPBRW: Dictionary
1429	RSPEC	HPBRW: Specification
1430	RSPCF	HPBRW: Specification (frozen)
1431	REXEC	HPBRW: Execution
1432	RJOB	HPBRW: Report Job
1433	ROUTI	HPBRW: Intermediate Report
1434	ROUTD	HPBRW: Dictionary Output
1435	PRINT	HPBRW: Print
1436	RCONF	HPBRW: Configuration
1437	RDICN	HPBRW: Dictionary

1438	REXNM	HPBRW: Execution (NM)
1441	PIF	reserved
1446	h1446	HPDESK: LISTKEEPER FILE
1447	h1447	HPDESK: FORM DATA FILE
1448	h1448	HPDESK: HPDESK CONFIG FILE
1449	h1449	HPDESK: SEARCH SPEC (LIB)
1450	h1450	HPDESK: ENCRYPTED FILE
1455	h1455	HPDESK: DESK PACKAGE
1461	NMOBJ	Native Mode Object
1462	PASLB	PASCAL Source Library
1476	TIFF	Tag Image Format
1477	RDF	Revisable Document Format
1478	SOF	Serial Object Format
1479	GPH	Chart for Charting Gallery
1480	GPD	Data for Charting Gallery
1483	VCGPM	Virtuoso Code Generator Processed Micro File
1484	FRMAT	Formatter
1485	DUMP	Dump
1486	NWMD0	New Wave Mail Distribution
1491	X4HDR	X.400 Header
1500	WP1	Other WP1
1501	WP2	Other WP2
1502	LO123	Lotus 123 spreadsheet
1514	FTCF	Form Tester Command Spec
1515	INSP	Spooled input
1516	OUTSP	Spooled output
1517	CHKSP	Checkpoint
1521	DSKIT	HPDESK Intrinsic Transaction
1526	MSACK	Mail Server Acknowledgment
1527	MSNDN	Mail Server Non-Delivery Notification
1528	MSTRC	Mail Server Trace
1560	h1560	HPDESK: MSWORD/Macintosh Document
1581	h1581	HPDESK: IBM DISPLAYWRITE
1587	ps	HPDESK: POSTSCRIPT WRITER
1601	h1601	HPDESK: TIFF DOCUMENT
1602	h1602	HPDESK: PCL DOCUMENT
1607	gif	gif
1616	h1616	HPDESK: WORDPERFECT Document
1617	h1617	HPDESK: WORDPERFECT Document
1618	h1618	HPDESK: MS WORD3.01/Macintosh
1619	h1619	HPDESK: MSWORD/PC Document
1620	h1620	HPDESK: MSWORD/WINDOWS Document

1621	h1621	HPDESK: SAMNA WORD Document
1622	h1622	HPDESK: AMIPRO Document
1626	xls1	HPDESK: MS-EXCEL Spreadsheet
1645	h1645	HPDESK: MACBINARY
1647	h1647	HPDESK: MS-POWERPOINT
1701	trek	STARTREK/SUPRTREK save game
1727	h1727	HPDESK: FREELANCE
1729	h1729	HPDESK: DELTAGRAPH
1731	h1731	HPDESK: EXPRESSIONIST
1734	h1734	HPDESK: MS-MAIL MSG DATA
1735	h1735	HPDESK: MS-WRITE Document
1745	h1745	HPDESK: MS-VIDEO
1746	h1746	HPDESK: MS-WINDOWS BITMAP
1850	h1850	HPDESK: PICT
1851	h1851	HPDESK: PICT FILE/Macintosh
1852	h1852	HPDESK: TIFF FILE/Macintosh
1853	h1853	HPDESK: EPSF
1854	h1854	HPDESK: EPSF FILE/Macintosh
1855	h1855	HPDESK: MACWRITE
1856	h1856	HPDESK: MACWRITE/Macintosh
1857	h1857	HPDESK: MACWRITE II
1858	h1858	HPDESK: MACWRITE II/Macintosh
1859	h1859	HPDESK: MACPAINT
1860	h1860	HPDESK: MACPAINT/Macintosh
1861	h1861	HPDESK: MACDRAW
1862	h1862	HPDESK: MACDRAW/Macintosh
1863	h1863	HPDESK: MACDRAW II
1864	h1864	HPDESK: MACDRAW II/Macintosh
1865	h1865	HPDESK: MACDRAW PRO
1866	h1866	HPDESK: MACDRAW PRO/Macintosh
1867	h1867	HPDESK: HYPERCARD
1868	h1868	HPDESK: HYPERCARD/Macintosh
1869	h1869	HPDESK: RESEDIT
1870	h1870	HPDESK: RESEDIT/Macintosh
1871	h1871	HPDESK: MS Word Macintosh 3.x
1872	h1872	HPDESK: MS-WORD/MAC/Macintosh
1873	h1873	HPDESK: MS-EXCEL/Macintosh Spreadsheet
1874	h1874	HPDESK: MS-EXCEL/Macintosh
1875	h1875	HPDESK: MS-EXCEL/Macintosh CHART
1876	h1876	HPDESK: MS-EXCEL/Macintosh CHART
1877	h1877	HPDESK: MS-EXCEL/Macintosh MACRO
1878	h1878	HPDESK: MS-EXCEL/Macintosh MACRO

1879	h1879	HPDESK: MS-EXCEL WORK
1880	h1880	HPDESK: MS-EXCEL WORK/Macintosh
1881	h1881	HPDESK: MS-EXCEL CHART
1882	h1882	HPDESK: MS-EXCEL CHART/Macintosh
1883	h1883	HPDESK: MS-EXCEL MACRO
1884	h1884	HPDESK: MS-EXCEL MACRO/Macintosh
1885	h1885	HPDESK: SUPERPAINT 2.x
1886	h1886	HPDESK: SUPERPAINT 2.x /Macintosh
1887	h1887	HPDESK: SUPERPAINT 3.x
1888	h1888	HPDESK: SUPERPAINT 3.x /Macintosh
1889	h1889	HPDESK: PAGEMAKER/Macintosh
1890	h1890	HPDESK: PAGEMAKER File/Macintosh
1891	h1891	HPDESK: WORDPERFECT/Macintosh Document
1892	h1892	HPDESK: WORDPERFECT/Macintosh
1893	h1893	HPDESK: LOTUS 1-2-3/Macintosh
1894	h1894	HPDESK: LOTUS 1-2-3/MACINTOSH
1895	h1895	HPDESK: PANORAMA II
1896	h1896	HPDESK: PANORAMA II/Macintosh
1897	h1897	HPDESK: FILEMAKER PRO
1898	h1898	HPDESK: FILEMAKER PRO/Macintosh
1900	h1900	HPDESK: GIF FILE/Macintosh
1901	h1901	HPDESK: OMNIS 7 APPL
1902	h1902	HPDESK: OMNIS 7 APPL/Macintosh
1903	h1903	HPDESK: OMNIS 7 DATA
1904	h1904	HPDESK: OMNIS 7 DATA/Macintosh
1905	h1905	HPDESK: QUARK XPRESS
1906	h1906	HPDESK: QUARK XPRESS/Macintosh
1907	h1907	HPDESK: CANVAS
1908	h1908	HPDESK: CANVAS/Macintosh
1909	h1909	HPDESK: DELTAGRAPH/Macintosh
1910	h1910	HPDESK: EXPRESSIONIST/Macintosh
1911	h1911	HPDESK: FREEHAND
1912	h1912	HPDESK: FREEHAND/Macintosh
1913	h1913	HPDESK: PERSUASION
1914	h1914	HPDESK: PERSUASION/Macintosh
1915	h1915	HPDESK: ILLUSTRATOR EPSF
1916	h1916	HPDESK: ILLUSTRATOR EPSF/Macintosh
1917	h1917	HPDESK: PHOTOSHOP
1918	h1918	HPDESK: PHOTOSHOP/Macintosh
1919	h1919	HPDESK: MS-POWERPOINT/Macintosh
1920	h1920	HPDESK: FrameMaker
1921	h1921	HPDESK: FrameMaker/Macintosh

1926	h1926	HPDESK: INTELLIDRAW
1927	h1927	HPDESK: INTELLIDRAW/Macintosh
1928	h1928	HPDESK: MINICAD+
1929	h1929	HPDESK: MINICAD+/Macintosh
1938	h1938	HPDESK: LOTUS 1-2-3 WIN V4
1939	h1939	HPDESK: QUICKTIME MOVIE
1940	h1940	HPDESK: QUICKTIME MOVIE/Macintosh
1941	h1941	HPDESK: MICROCAL ORIGIN
1942	h1942	HPDESK: LOTUS ORGANISER
1944	h1944	HPDESK: ORG PLUS FOR WIN
1945	h1945	HPDESK: HARVARD GRAPHICS
2057	h2057	HPDESK: MS-WINDOWS SOUND
2065	h2065	HPDESK: MS-WINDOWS CONFIG
2066	h2066	HPDESK: MS-WINDOWS CARD
2067	h2067	HPDESK: MS-WINDOWS HELP
2068	h2068	HPDESK: MS-WINDOWS ICON
2500	HPWR	HP OpenDeskWriter Backup
2501	STORE	TurboSTORE/iX + store-to-disc
2502	STDIR	TurboSTORE/iX on-disk directory of files stored
3333	hpwr	HPWORD scratch file
5000	dbg66	Encapsulation file for option 66 of DBGeneral
5100	pdca	PDCMAN data
5101	tpdsk	TAPEDISK data
5102	redo	REDO file
7210	oalep	OAL Compiled Procedure
7350	ovini	OVINIT (Omnilink security file)
7604	odxfr	ODXTRANSFER
7679	disc1	DISC internal permanent file
7680	disc2	DISC internal permanent file
7681	disc3	DISC internal permanent file
7682	Disc4	DISC internal permanent file
7684	Oeip	Omnidex external index pointer file
7690	Owcfg	OmniWindow configuration
8501	Mmch	Multiview & ManMan compiled help
8502	Mvglr	Multiview General Ledger Report
21074	Rrvtn	RoadRunner virtual tape file (non-priv)

Appendix C – Regular Expressions

What Are Regular Expressions?

Server version 4.8 or higher is required.

Regular expressions might look like wildcards used in the **Pattern** search option. They are sometimes compared to wildcards but, in fact, they are much more powerful. They can also be much more complex. You have to practice in order to use them efficiently and to their full potential. For brevity, we will often refer to regular expressions simply as regexp.

Although all regexp implementations share a basic set of metacharacters and syntax rules, other tools and programs might have different extensions and variations. Be aware that the version used in Qedit might not have all the extensions you are currently familiar with. For example, Qedit does not support alternation with the vertical bar “|” (equivalent to a logical “or”). Also, this is the first implementation in Robelle products. We will be looking at other tools and might decide to extend our own implementation in future releases.

If you are interested in learning more about regular expressions, you should get a copy of *Mastering Regular Expressions* written by Jeffrey E. F. Friedl and published at O'Reilly & Associates, Inc. This book covers most regular expression implementations, the differences between each one, how most regexp engines work and some tips on how to get the best performance from each type.

Metacharacters

Qedit supports the following metacharacters:

- ^ Start-of-line anchor
- \$ End-of-line anchor
- .
- ?
- Matches zero or more of the preceding character
- +
- [Start a character class
-] End a character class
- ^ If 1st character in character class, negates class

- (Subpattern start
-) Subpattern end

Anchor Characters

In general, a regexp can find a match anywhere in the text as long as they do not span multiples lines. There are 2 exceptions to this rule. The start-of-line "^" and the end-of-line "\$" anchors. They are called anchors for very good reasons.

These actually indicate that the match must occur at fixed positions within the line. The start-of-line anchor forces the string to be at the very beginning. If you enter

```
^abc
```

the line will be selected only if the string *abc* is the first thing on the line. Thus,

```
abcdefghij Will be selected
xyzabc Will not be selected
```

Similarly, the end-of-line anchor specifies that the string must appear as the last thing on the line.

```
abc$
```

In this example, the lines must end with the string *abc*. There must not be anything else, even spaces, after it.

```
abcdef Will not be selected
xyzabc Will be selected
```

You can combine the anchors to verify that lines contain only a certain string and nothing else. Simply use

```
^abc$
```

Each line has a start and an end of line. So, if you search for “^” or “\$”, Qedit finds a match on the previous or next line. The only difference is the cursor position. A successful match with a start-of-line anchor places the cursor at the beginning of the line. A match with the end-of-line anchor places the cursor after the last character on the line. Also note that, since these do not represent actual characters, the result of a search is an empty selection.

If the anchor characters are used anywhere else, they lose their metacharacter status and become ordinary characters.

Remember that COBOL tags, when displayed, are part of the text. Thus, if you are trying to search for a regexp at the end of the line, you have to code the regexp accordingly or hide the tags while doing the search.

Match Any Character

The period, or dot, is used to match any character. The character can be of any type. As long as there is something in that position, there will be a match. For example,

```
abc.xyz
```

selects any line that contains the strings *abc* and *xyz* separated by a single character. That character can be anything e.g. *l*, *w*, *#*, etc.

Optional Character

You can check the absence or presence of a character by suffixing it with a question mark. In a regexp, this indicates that the preceding character might not be there. If it is present, it must be there only once.

```
ab?c  Matches ac or abc only
```

Repeating Characters

There are different ways you can check for repetition of characters. If there is potential for a character to appear more than once, you can use the asterisk or plus sign quantifiers. The quantifier is applied to the character to its immediate left.

There is a very small difference between the two quantifiers. The asterisk represents **zero** or more occurrences of the preceding character. That is the character is optional but, if it is there, it can repeat multiple times.

The plus sign represents **one** or more occurrences. This means the character must appear at least once but it can appear multiple times.

```
ab*c  Matches ac, abc, abbc, and so on  
ab+c  Matches abc, abbc but not ac
```

Character Class

When you have to check for a fixed string of characters, it is easy enough to simply type it at the actual regexp. Entering

```
abc123
```

only finds exact matches. What if you want to find the string *abc* followed by a numeric digit? There are no specific metacharacters for digits or alphabetic characters. However, regular expressions have a concept called *character class* to address these issues. Actually, character class is a lot more powerful and flexible than metacharacters for specific types of text.

A character class is enclosed between square brackets. Beware that most regexp metacharacters in the list above lose their meanings inside a character class. The start-of-line anchor acquires a different definition and a new metacharacter, hyphen appears.

A character class is a list of possible values for that position in the string. The character class can be as long as needed. A character class for numeric digits would be:

```
[0123456789]
```

Note, the list does not have to be in sorted order. You could have entered the digits in reverse order or in random order and the character class would still be valid. It just makes it harder to verify that all digits are there. Similarly, a character class for lowercase letters would be:

```
[abcdefghijklmnopqrstuvwxyz]
```

It is really important to understand that a match occurs if **one** of the characters in the class is found. Using the *abc* example above, if we wanted to make sure we had this string followed by a digit, we would enter:

```
abc[0123456789]  Matches abc0, abc1, and so on to  
abc9
```

To make it even more restrictive, we could have used:

```
abc[13579] Matches abc followed by one odd digit
```

Because a character class is only a list of possible values, you can mix and match all the characters in the ASCII code table.

```
p[imy246!.*]e Matches pie, pre, p4e, p*e
```

This would find text that starts with the letter *p* and ends with an *e* with a single character between them, matching one of the letters *a*, *m* or *y*, one of the digits *2*, *4* or *6*, an exclamation mark, a period or an asterisk. Note the period and asterisk are not metacharacters anymore.

Of course, if the character class contains a lot of possible values, it can be tedious and error-prone to enter each character. The hyphen can be used as a range indicator. It is a character-class metacharacter. Simply specify the first and last characters in the range. Numeric digits could then be coded as

```
[0-9]
```

Lowercase letters could be coded as

```
[a-z]
```

You can also combine ranges with single values as in:

```
abc[0-9a-z!.*]
```

The hyphen is interpreted as a range indicator only if it is between two other characters. If it is the first or last character in the class, it is used as a literal.

Negated Character Class

The caret "^" takes on a different meaning inside a character class. It used as a literal anywhere in the class except if it is listed first. When entered immediately after the left square bracket, it negates the whole class. This means a match is found if the text **does not** contain any of the characters in the class.

```
p[246^]e Matches p2e, p^e and so on  
p[^246]e Matches pae, p3e, and others
```

In the last example, the caret negates the *2*, *4* and *6*. The regexp is true if the text starts with the letter *p* and ends with the letter *e* and there is a single character between them that is not *2*, *4* and *6*.

Repeating Character Class

Because a character class is interpreted as a single character, you can use the optional metacharacter, question mark, and quantifier metacharacters, asterisk and plus sign, to further qualify a character class. For example, if we wanted to allow one or more numeric digits after the *abc* string, we could use the following regexp:

```
abc[0-9]+
```

Escape Character

Other characters used in a regular expression might also have special meanings. The most important one is probably the escape character. In Qedit, the backslash is the escape character. A metacharacter loses its special meaning if preceded by a backslash. In the example,

```
abc [123]
```

square brackets indicate a character class. This regexp would match *abc1*, *abc2* or *abc3*. If we escape the square brackets as in

```
abc&\[123&\]
```

the square brackets are then used as literals. This means they are now part of the string. The only matching value is then *abc[123]*.

If you want to search for a backslash, simply enter two in a row e.g. `\\`. The only exception to this rule is the start-of-line anchor “`^`”. Because “`^`” is also an escaped sequence, there is only one way to search for the literal “`^`”. You have to use the hexadecimal value of the character. You would use the following string in your regular expression:

```
\\x05e
```

Escaped Sequences in Regular Expressions

Additionally, the escape character can be combined with other characters to represent other non-printing characters. Qedit recognizes the following escaped characters. These should not be confused with escape sequences used to control the display on HP-type terminals. These are not metacharacters either.

<code>\\b</code>	Backspace
<code>\\e</code>	ASCII escape character (ESC)
<code>\\f</code>	Formfeed
<code>\\n</code>	Newline (linefeed)
<code>\\r</code>	Carriage return
<code>\\s</code>	Space
<code>\\t</code>	Horizontal tab
<code>\\DDD</code>	1-3 octal digits representing a character's ASCII value
<code>\\xDDD</code>	1-3 hex digits representing a character's ASCII value
<code>\\^C</code>	Control code e.g. Control-G (^G) is the bell character

For example, you would use:

<code>\\t</code>	All lines with a tab character
<code>\\e&d@</code>	Terminal escape sequence ESC&d@

Backreferences in Regular Expressions

We have seen basic expressions where almost everything evolves around single characters of text. Even character class lists are really used to match a single position.

You can use parentheses to divide a long regular expression into smaller portions. Each portion then becomes a regexp of their own. This does not affect the way a string search is done. However, each subpattern can then be used in a replacement operation.

Subpatterns are numbered from 0 to 9. Subpattern 0 is reserved and represents the complete matched string. Note that subpattern 0 is implicit and is always available, even if the expression does not contain any parentheses. Explicit subpatterns are numbered from 1 to 9 starting from the left of the expression.

Subpatterns can be referenced in a replacement regexp using the escape character, backslash, followed by the number. When applying the replacement string, backreferences are substituted with the actual matching text. Backreferences can be used as many times as needed. Each reference would end up with the original text.

Let's say we have a file that contains a series of phone numbers. In North America (and possibly other parts of the world), phone numbers contain a 3-digit area code followed by a 3-digit exchange number and, finally, a 4-digit individual number.

Unfortunately, in our case, the phone numbers are just series of 10 numeric digits without separators. For example,

```
1234567890
1112224444
9087374456
```

We would like a fast and easy way to format them so the numbers are easier to read. To find all these strings, we can use the following regexp:

```
([0-9][0-9][0-9])([0-9][0-9][0-9])([0-9][0-9][0-9][0-9])
```

We use the `[0-9]` character class to specify that we are expecting numeric

digits only. There are 3 subpatterns in the regexp. Each subpattern is enclosed in a set of parentheses. The first subpattern repeats the numeric character class 3 times. It represents the digits that are part of the area code. This is subpattern `\1`.

The second subpattern also has the character class repeated 3 times. These represents the exchange number. This is subpattern `\2`.

The third and last subpattern uses the character class repeated 4 times. These represents the individual number. This is subpattern `\3`.

Subpattern `\0` represents all 10 digits.

To reformat, we can now combine backreferences with other characters to format the numbers anyway we like. Let's say we want to put the area code in parentheses, insert a space after it and insert a dash "-" between the exchange number and the individual number. We would use the following substitution string:

```
(\1) \2-\3
```

The list above would then look like:

```
(123) 456-7890
(111) 222-4444
(908) 737-4456
```

You can use subexpressions to reorder the text in lines. For example, if we wanted to reverse the telephone numbers showing the last four digits first, then the first three digits of the telephone number, followed by the area code, we could use the following substitution string:

```
Last: \3 Middle: \2 Area: \1 Complete: \0
```

Which would then produce the following (assuming that we started with the same data in this example and used the same regexp):

Last: 7890 Middle: 456 Area: 123 Complete: 1234567890
Last: 4444 Middle: 222 Area: 111 Complete: 1112224444
Last: 4456 Middle: 737 Area: 908 Complete: 9087374456

Appendix D - Language Templates

Overview

In order to take advantage of the syntax coloring feature of Qedit, the syntax of a language must be defined in a file known as template. Language templates are regular text files containing information formatted according XML rules. For that reason, template files must have a `.xml` extension.

Template Location

Robelle-supplied language templates are stored in the `system\language` subdirectory below the Qedit base directory e.g. `\robelle\qedit`. These files should not be modified as they might be overwritten when a new version is installed.

If you wish to create new templates or modify one of Robelle's templates, you should create them in `\user\language`. Typically, the `user` subdirectory contains files that have been customized and should be shared by all Qedit users. The `system` and `user` directories are always scanned.

You can create your own language templates and store them the location of your choice. You simply have to specify the location on the **General** page of the **Preferences** dialog box.

When Qedit starts up, it scans all the templates found at the above locations and loads the definitions into internal structures. These structures do not change after that point. To use a new or updated language template, you have to exit out of Qedit and start it up again.

Here are the language templates supplied by Robelle:

- `allbase.xml` (Allbase SQL)
- `c.xml` (C and C++)
- `cobol.xml` (Cobol, Cobfree)
- `cobx.xml` (CobolX)
- `fortran77.xml`

- `html.xml`
- `java.xml`
- `job.xml` (MPE commands)
- `MySQL.xml` (MySQL SQL)
- `oracle.xml` (Oracle SQL)
- `pascalix.xml`
- `powerhouse.xml`
- `qsl.xml`
- `spl.xml`
- `xml.xml`

Which Language?

See the host-based Qedit User Manual for details

Qedit uses different methods to determine the language of a file. For local files and HP-UX files, syntax coloring uses the file extension to determine its language.

When you open an MPE host file, the MPE server looks at physical characteristics of the file and its filecode. If it can, the server assigns a language based on that information. If the file is in the POSIX namespace, the server also looks at the file extension.

The server communicates the language information to the client. When you open a file for the first time, the client uses the server information and stores it in the document database. The client relies and uses the document database from that point.

You can change the language in the client at any time. The new information is saved in the document database. This means that opening a file in host-based Qedit might get a different language than opening it in Qedit for Windows. It also means that the client always uses the language that you defined last time you accessed the file.

Language Template Format

Template files use XML syntax. Here are some general rules used in the templates:

- Description of the language is done using `<starttag>...</endtag>` blocks.
- Blocks can be nested. In other words, a block can contain other `<starttag>...</endtag>` blocks.
- Tags can contain optional attributes.
- Attributes are delimited by one or more spaces.
- Attribute values must be enclosed in double-quotes.
- To avoid confusion and minimize parsing errors, enter delimiter characters with their ASCII codes using `igits`; syntax where *digits* are hexadecimal.
- Comments in the language template file itself are identified by `<!--` at the beginning and `-->` at the end.

- Unless specified otherwise, tags, attributes and values are not case-sensitive.

Language Block

The **Language** block is required. It's the main block which contains all other blocks describing the language. The **Language** tag has the following attributes:

- **Name:** name of the language. If two language templates have the same **Name**, the last template encountered is used for coloring. The name is optional and there is no default value.
- **Case:** indicates whether the language tokens are case-sensitive or not. If the language is case-sensitive, enter the word `significant`. Anything else means not case-sensitive. The attribute is optional and the default is not case-sensitive.
- **QeditLanguageCode:** list of numeric codes where each code is a number corresponding to a Qedit Language. There can be as many codes as needed. This is useful for language codes that basically identify the same language such as C and C++ or Cobol, Cobolx and Cobfree. Language codes are optional and there are no default values.

The same language code should not be used in more than one template. If this happens, Qedit uses the first one it encounters starting from the end of the list. For example, if the HTML template has the same language code as XML, the definition from XML is used. Valid codes are:

Language Code	Description
0	HP's Systems Programming Language (SPL)
1	Fortran
2	Cobol
4	Cobol with comments
5	Job stream
6	RPG
7	Free-format text up to 256 bytes
8	Pascal
9	Free-format text up to 8,172 bytes
10	C language
11	C++ language
12	Powerhouse
13	Pascal with long lines
14	Free-format Cobol
15	HTML
16	XML
17	Java
18	Qedit Scripting Language (QSL)

- **StartColumn:** column number where text starts on each line. This attribute is optional and must be greater than 0. The default is 1.

- **Horizon**: maximum number of lines to scan to find the start of a class e.g. multi-line comments. It is optional and must be greater than 0. The default is 100.

For example, the following tag identifies the SPL language which is not case-sensitive, lines start in column 1 and the language code is 0.

```
<language name="SPL" case="notsignificant" startcolumn="1"
qeditlanguagecode="0">
  language specifications
</language>
```

For each language, there are multiple blocks describing individual elements of the language, making up the language specifications. All these blocks must be define between the `<language>` and `</language>` tags.

File Extensions

The **Extension** blocks describe file extensions used to identify the language of a file. Extension blocks are optional. Each block contains one extension string. Extensions can be as long as needed and can be any character string. Extensions are not case-sensitive. In other words, ".c" is the same as ".C".

Extensions are applicable to local, HP-UX and Posix files. They are sort of dynamic. If you add an extension to a language template, simply restart Qedit for Windows to implement it immediately.

Extension support for HP-UX and Posix files is done a little differently. The server always assign a language code to a file. The language code is checked against the language templates. If there's a match, Qedit uses that language. If there's no match and the language code is Text or Data, which are generic types, Qedit looks at the file extension. If there's a match, that's the language. If the language code and extension do not match a template, the file simply does not have a language and no syntax coloring is done.

Here are file extensions identifying C or C++ files:

```
<extension>h</extension>
<extension>c</extension>
<extension>cpp</extension>
```

String Delimiters

The **Quote** blocks define characters that delimit literals within the language. Quote blocks are optional. Each **Quote** block contains a **StringDelimiter** block and optional **Escape** blocks.

The **Quote** tag attributes are:

- **Type**: indicates if a string started with this delimiter can span multiple lines. If strings can be continued on separate lines, use `Continuable`. Anything else means the string ends with the current line. It is optional and the default is not continuable.
- **ReopenContinuation**: indicates whether an opening quote character is required on the continuation lines. If required, enter `true`. Anything else means the delimiter is not required on continuation lines. It is optional and the default is not required.

There must be one and only one **StringDelimiter** block. Each block defines a delimiter. A delimiter must be a single character. The **StringDelimiter** tag has the following attribute:

- **Double**: indicates that two consecutive delimiters mean the delimiter is escaped i.e. it is used as is. Enter `escape` if that's the case. Anything else means consecutive delimiters have no special meaning. It is optional and the default is not escaped.
- The **Escape** block defines an escape character. If the delimiter is preceded by an escape character, the delimiter is used at face value and not as a closing delimiter. **Escape** blocks are optional but, if specified, there can be up to 10 escape entries and each one can not have more than 7 characters.

This example defines the double-quote (ASCII hex 22) as a delimiter. A string defined with it can span multiple lines. To escape a double-quote i.e. not interpret it as the start or end of a string, it can be preceded by a backslash or two question marks.

```
<quote type="continuable">
<stringdelimiter>&#x22</stringdelimiter>
<escape>\</escape>
<escape>??</escape>
</quote>
```

Continuation Schemes

The **Continuation** block describes how to identify continuation lines i.e. language statements that span multiple lines. This block is optional but, if specified, there must be only one. Each block can contain **Termination** or **Positional** blocks.

There can be a mix of **Termination** and **Positional** blocks and each continuation identifier must have only 1 character.

A **Termination** block indicates that the continuation character should be the last character on the line i.e. the line is going to be continued. A continuation character is required for a **Termination** block.

A **Positional** block indicates that the continuation character has to be at a specific location on the line. This block has only one attribute:

- **Position**: is the column number where continuation character should be. The position is required and must be greater than 0.

The continuation character is optional for this type of continuation. If specified, a continuation line is assumed only if this particular character is used at the defined position. If no continuation character is specified then any character found at that position is presumed to be a continuation.

For example, this continuation block defines the backslash as an end-of-line continuation character. It also defines the hyphen in column 6 as a continuation.

```
<continuation>
<termination>\</termination>
<positional position="6">-</positional>
</continuation>
```

Comment Schemes

The **Comments** block defines characters or strings used to identify comment text within the file. A **Comments** block is optional but, if specified, there must be only

one. The block can contain **Endline**, **Inline** and **LineDesignator** blocks. There can be as many of these blocks as needed. Comment identifiers in these blocks can not have more than 15 characters each.

Endline blocks define identifiers that makes the rest of the current line a comment. There can be as many **Endline** blocks as needed.

Inline blocks define identifiers marking the start and end of a comment. The beginning-of-comment identifier is defined with a **BeginComment** block. There must be only one of these per **Inline** block.

The end-of-comment identifier is defined with a **EndComment** block. There must be only one of these per **Inline** block. In a file, anything between the start-of-comment and end-of-comment identifiers is considered a comment and can span multiple lines. There can be as many **Inline** blocks as needed.

An **Inline** block can have a **Type** attribute. If the value is **Keyword**, it means the **BeginComment** and **EndComment** must appear as words to be recognized. For example, if the **BeginComment** string is `comment`, "comment" starts a multiline comment but "commentary" and "comments" do not.

A **LineDesignator** block indicates that the comment character has to be in a fixed position within a line. This block has one attribute:

Position: column number where the comment character must appear. It is required and must be greater than 0.

The comment identifier is optional. If specified, a comment is assumed only if this particular identifier is used at the defined location. If no identifier is specified then any character found at that position is presumed to be a comment.

In this example, two slashes identifies end-of-line comments. In other words, whenever two slashes are encountered, whatever text follows them is considered a comment. A slash-asterisk sequence starts a comment section. An asterisk-slash sequence ends it. A dollar-sign in column 1 indicates that the current line is a comment.

```
<comments>
<endline>>//</endline>
<inline>
<begincomment>/*</begincomment> {/* starts comments}
<endcomment>*</endcomment> {*/ ends comments }
</inline>
<linedesignator position="1">${</linedesignator>
</comments>
```

Token Character Set

The **TokenCharset** block defines all the characters that can be used in user-defined words such as variable names. This block is optional. If specified, it should contain a list of valid characters. The list is similar to a character class in a regular expression. Character ranges can be specified by entering the first and last character of the range separated by a dash e.g. "a-z" means all lowercase letters between "a" and "z". The character list can be enclosed in square brackets, "[" and "]". If so, the brackets are discarded.

In this example, user-defined words can have upper and lowercase letters, digits 0 through 9 and the underscore character.

```
<tokencharset>[A-Za-z0-9_]*</tokencharset>
```

Keywords

The Keyword blocks are used to specify all reserved words for the corresponding language. Each block defines one keyword. There can be as many as needed and each one can be as long as needed.

In this short example, we define four keywords: While, Break, Move and Perform.

```
<keyword>while</keyword>  
<keyword>break</keyword>  
<keyword>move</keyword>  
<keyword>perform</keyword>
```


Glossary of Terms

Access keys

Underlined character in menu and command names. Combined with the ALT key, they allow quick access to functions without using the mouse or the cursor keys. They are sometimes called mnemonics.

Active

The active document window displays the file in which the changes are going to be applied.

Caret

Character indicating the location for the next operation. Its functionality is equivalent to the insertion point. A caret is usually represented by a character resembling a lowercase V, upside down.

CI

See Command interpreter

Command interpreter

(CI) On MPE hosts, the program that is automatically initiated in a session or job. The CI accepts, interprets and executes the operating system commands on your behalf. It is equivalent to a shell on a UNIX host.

Current working directory

(CWD) Term used for UNIX host and POSIX namespace on MPE to identify the directory in which most file operations are occurring. Qedit looks for files to open or creates new files in the CWD unless the file name is qualified.

CWD

See Current Working Directory

DNS

See Domain Name server

Domain Name Server

(DNS) Software used to cross-reference IP addresses to host names.

End-of-line character

Operating systems have different ways of identifying the end of a line in a file. Some use a LINE FEED or NEWLINE character, a CARRIAGE RETURN or a combination of a CARRIAGE RETURN/LINE FEED.

Firewall

Security device (hardware or software) controlling access to a network. Usually controls which IP addresses are allowed to access specific port numbers and services.

Folder

Sometimes called a directory. Item used as a container for related folders or documents.

HFS

See Hierarchical File System.

Hierarchical File System

(HFS) Sometimes called POSIX namespace on MPE. File system in which files and directories are arranged in hierarchical fashion. There is a root directory from which all other components branch out.

Insertion point

Exact location within the file in which the edit operation is going to take place. Usually indicated by a blinking cursor. If you select some text, the insertion point is the highlighted text itself.

Internet Service Provider

(ISP) Companies offering Internet access to individuals or organizations. They usually offer services such as electronic mail, World Wide Web hosting, etc.

IP address

Internet Protocol address. Unique identifier required for all computer systems connected to a network. The address is made up of a series of 4 numbers separated by a period. Each number can have a value between 1 and 255. e.g., 128.11.200.77

NOTE: Do not enter leading zeros in any part of an IP address. A leading zero indicates that the value is specified in octal notation. In the example above, if you had entered 128.011.200.077, the IP address would have been interpreted as 128.9.200.63.

ISP

See Internet Service Provider

Mnemonic

Short series of characters used to represent an attribute, a concept or provide fast access to a function. They are called mnemonics as they are typically easy to remember. See also Access Keys

Most recently used

(MRU) List that contains information entered most recently in a specific area. The file name list box of the **Open Host File** dialog box and the **Search for** list box of the **Find String** dialog box are examples of MRU lists.

MRU

See Most Recently Used

Native Language Support

(NLS) Subsystem that allows proper handling (conversion, formatting and collation sequence) of national characters. Refer to documentation from HP for details.

NLS

See Native Language Support

Pane

When using a split view, sections of the file on either side of a split bar.

PID

See Process IDentification number

POSIX

Hierarchical File System on HP e3000 running MPE/iX 4.5 or later. Files and directories created under the HFS follow the same rules as other machines running UNIX.

Process Identification number

(PID) Server processes usually have a unique ID number assigned by the operating system. There is one server process per connection. The PID can be used to scan the log files and find out what was done during that editing session.

Qedit Scripting Language

(QSL) Scripting language offering access to Qedit for Windows functions and objects. It can be used to customize the Qedit client environment, perform common editing tasks, define new commands to perform custom functions or automate repetitive tasks for hands-off operations. Refer to the *QSL User Manual* for details.

QSL

See Qedit Scripting Language

Record length

Maximum number of characters that can fit on a line within the file. Often related to the Language of the file.

Remote Process Management

(RPM) Server process running under MPE/iX allowing the creation of sessions and processes from remote computers.

RPM

See Remote Process Management

Scroll bar

Area on the right hand side and at the bottom of a window that allows you to move around in a file to display different sections. A scroll bar includes a scroll arrow at both ends and a scroll box.

Shell

On UNIX hosts, this is a program that accepts, interprets and executes commands that you normally enter in a session or script. It is equivalent to the command interpreter on MPE.

Split bar

Yellow line displayed between two panes when using a split view. The split bar can be moved to change the size of a pane.

Split box

Red area that appears at the top of a vertical scroll bar and on the left of an horizontal scroll bar. The split box can be used to split a document window into two or more panes. This allows you to look at different sections of the same file.

UDC

See User Defined Commands

User Defined Command

(UDC) MPE on HP e3000 computers allows you to create your own set of commands. These new commands can execute one or more other commands (MPE, command files or other UDCs).

XML

Stands for eXtensible Markup Language. Very similar to the popular HTML (HyperText Markup Language)

Index

Symbols

- ! (exclamation point) search pattern 143
- # (number sign) search pattern 143
- \$ (dollar sign) metacharacter 251
- & (ampersand) search pattern 143
- ((left parenthesis) metacharacter 252
-) (right parenthesis) metacharacter 252
- * (asterisk) metacharacter 251
- . (dot) metacharacter 251
- ? (question mark) metacharacter 251
- ? (question mark) search pattern 143
- @ (at-sign) search pattern 143
- [(left square bracket) metacharacter 251
-] (right square bracket) metacharacter 251
- ^ (circumflex) character class negation 252
- ^ (circumflex) metacharacter 251
- ^ (circumflex) search pattern 143
- ~ (tilde) search pattern 143
- + (plus sign) metacharacter 251

Dollar-Oldpass

- \$Oldpass
 - host command 172

A

- About Qedit dialog 207
- Access log file 217, 218, 220
- Access mode 34
- Acrobat Reader 11
- Active connections 28
- Add scripts 177
- Adobe Acrobat 11
- Allbase
 - syntax coloring 123
- Anchor characters 252
- Attributes
 - document 75

- Auto-indent 68, 118
 - default 125

B

- Backreferences, regexp 256
- Backspace key 91, 100
- Binary data 82
- Braces
 - matching 122
- Brackets
 - matching 122
- busy workfile 194, 224
- BYE, MPE command 173

C

- Cache 97, 214
- Cancel button 33, 34
- Caret 14, 148
- Carriage-return terminator 68
- Caseless search 142
- Character class 253
- Character set 56
- Characters
 - display nonprinting 93
 - displaying invisible 121
 - international 51, 94
 - nonprinting 82, 91, 92
 - r_ansi font 93
 - special 51, 91, 92, 189
- Checked commands 25
- Clipboard 96
 - local 98
 - tab characters 104
- Close
 - forced 165
- COB74XL command 183
- COB85XL command 183
- COBOL 64, 113
 - tag 133, 134, 214
 - automatic 137
 - manual edit 133, 137
- COBOL compile, MPE 172
- COBOL copylib 190
- Color
 - adding scheme 128
 - changing scheme 128
 - connection default 56
 - defaults hierarchy 129
 - Local default 87
 - MPE default 87
 - removing scheme 129
 - schemes 125
 - UNIX default 87

- Column
 - delete 101
 - find in 145
 - insert 102
 - shift 102
- Command line interface 75
- Compile MPE source 183
- Configuration
 - server 19, 159, 227
- Connection 2, 53, 54, 161, 193
 - copying 58
 - creating 54
 - disconnect 53, 61, 164
 - information 212
 - maintaining 60
 - MPE host 56, 215
 - removing 61
 - renaming 61
 - Temporary 54
 - UNIX host 210–23, 220
 - using 29, 59, 80
- Connection List dialog 55, 58, 60, 61, 215
- Connection template 21, 56
- Controlling scripts 178
- Convert
 - text to workfile 154
 - workfile to text 155
- Copy Special dialog 98
- Copylib 190
- CR terminator 68
- Cursor 35
 - keys 31
- Customizing
 - client 13
 - connection 61
 - opening a file 87
 - save and close 165
 - search 148
 - server 19

D

- Debug
 - permanent settings 210
 - temporary settings 210
- Debug Controls dialog 208
- Delete column 101
- Detab 104
- Dialog
 - About Qedit 207
 - Connection List 55, 58, 60, 61, 215
 - Copy Special 98
 - Debug Controls 208
 - Directory 1, 29, 81
 - Expiry message 21

- File Options 212
- Go To Line 145
- Host command 168
- Login 222
- Manage scripts 177
- New Host File 29, 59, 63
- Open 46, 48, 64, 77
- Open Host File 29, 80
- Paste progress 97
- Preferences 13, 162, 232
- Replace 147
- Save As 163
- Script control 178
- Tip of the Day 22
- Directory 62, 221, 224
 - changing 73
 - creating 64
 - default HP-UX 58, 80, 220
 - dialog 29
 - Font 15, 112
 - layout 71
 - MPE vs POSIX 74
 - print 74
 - saving from 74
 - select files 74
 - sorting 72
 - wildcards 72
- Directory dialog 1, 81
- Disable host command 171
- Disconnect 53, 61, 164
- DNS 215, 218, 220–22, 220–22
- Document
 - attributes 75
 - font 111
 - information 211
- Document database 75
 - recovery 229
- Document type 33
- DOS files 68
- Drag-and-drop open 50, 51

E

- Embedded file 189
- Embedded files 76
- end-of-line characters 121
- End-of-line characters 43
- Entab 104
- Environment
 - MPE 171, 173
- Error
 - log file 217, 218, 220
 - messages 231
- Escape character, regexp 255
- Escaped sequence, regexp 255

- European currency 56
- Exchange position 141
- Executing host command 167
- Expiring version 21
- Expiry message dialog 21

F

- Favorites 83
 - add 83
 - add to folder 85
 - new folder 84
 - remove 86
- File 152, 223
 - autosave 166
 - closing 40, 164, 165
 - colors 66
 - converting 154
 - copying 151
 - creating 28, 31, 63, 64
 - defaults 87
 - DOS terminator 68
 - drag-and-drop open 50, 51
 - embedded 76, 189
 - filters 78
 - font 66
 - fully-qualified name 189
 - host 63, 79, 152, 164, 193
 - host printing 158
 - Information 210
 - line termination 68
 - local 28, 31, 64, 77, 163, 164, 225
 - local printing 155
 - Macintosh terminator 68
 - Most recently used 80
 - Most Recently Used 77
 - MRU 80
 - multiple local 79
 - names 32
 - opening 28, 77, 152
 - Options 65
 - quick open 95
 - recovery 223, 224
 - registering QSL 50
 - registering type 49
 - rename 153
 - saving 28, 161, 163
 - support 11
 - timestamp 166
 - types 33, 237
 - Unix terminator 68
 - unnumbered 81
- File as host command 170
- File extension
 - syntax coloring 123

- File labels, MPE 191
- File Options dialog 212
- File types 66, 73, 237
- Filename 189
- Filters 78
- Find 141–44, 141–44
 - first 144
 - in columns 145
 - in selection 145
 - last 144
 - options 144
 - part of file 145
 - selected text 144
- Find String dialog 142
- Firewall 1, 3, 193, 217
 - protocol 56, 217
- Fold lines 81
- Folder
 - Favorites 85
- Font 111
 - default 124
 - nonprinting characters 93
 - print 157
 - r_ansi 93
 - window size 40
- Force close 165
- FORTRAN command 183
- ftp 215, 219
- Function keys 204, 206

G

- Go To Line
 - dialog 145
- Goto
 - previous position 141
- Group 57, 80, 215, 219, 223
- Guide line 113, 118

H

- Hardware and Software Requirements 1
- HELLO, MPE command 173
- Help 4
- Highlights 4
- Host
 - printer 159
- Host command 167
 - \$Oldpass 172
 - BYE 173
 - clear log 169
 - COBOL compile 172
 - copy log 169
 - dialog 168
 - disable 171

- duplicate output 173
- environment 172
- HELLO 173
- interrupt 169
- logon UDC 172
- MPE 171
- nonprinting characters 168
- QEDSERVMODE 172
- server configuration 172
- settings 172
- UDC 172
- Host file 24, 193
 - closing 165
 - creating 63
 - MPE 212
 - opening 79, 152
 - wildcards 81
- Host line numbers 156
- Host name 46, 54–61, 54–61, 213, 215, 218, 220–22, 220–22
- Host printing 158
- HP 9000 1, 3, 53, 57, 217, 220
- HTML Editor 79

I

- Include files 180
- Indent 68, 118
 - default 125
- Information
 - connection 212
 - document 211
 - transmission 213
- Insert column 102
- Insert mode 90
- Insertion point 31, 141, 148
 - prevent sliding 148
 - undefined area 148
- Install 7
 - HP-UX server 220
 - Network 8
 - Server 9
 - Typical 7
 - Workstation 8
- Intelligent mouse 140
- International characters 51, 94
- Internet Explorer 79
- Internet resources 4
- Invisible characters 121
 - default 125
- IP address 55, 193, 215–20, 222, 232
- ISO8859 56

K

- Keepalive transaction 194
- Keyboard 90, 96, 141, 204
 - non-english 94

L

- Language 64, 66
 - location 12
- Language code
 - syntax coloring 123
- length
 - exceeded 82
- LF terminator 68
- Line length 64, 66
 - maximum 137
- Line number 67, 145–46
 - host 131
- line number, out of 83
- Line termination 68
 - default 137
- LineFeed terminator 68
- List \$include 180
- List all lines 180
- Listfile 73
- Load scripts 177
- Local clipboard 98
- Local file 154
 - closing 164
 - creating 31, 64
 - opening 77
 - operation 225
 - saving 163
- Local trace 208
- Log file
 - server 212, 217, 218, 220, 231, 270
- Login dialog 222
- Logon UDC 20, 172
- Long file name 49, 52

M

- Macintosh files 68
- Manage Scripts dialog 177
- Manuals 11
- Mark a position 141
- Maximum length 81
- Menu 45
 - application control 23
 - Edit 52, 65, 98, 100, 108, 146, 154
 - File 33, 54, 59, 61, 63, 64, 80, 108, 152, 155, 164, 188, 210, 232
 - Mark 145, 146
 - Options 13, 54, 58, 60, 61, 216, 218, 219, 221

- shortcut 46
- View 26, 30, 49, 50
- Window 87, 185
- window control 32, 40
- Menu bar 24, 45
- Messages 31
- Metacharacter 251
- Most recent files 77
- Mouse click
 - double 37, 40, 41, 42, 52, 66, 224
 - triple 41, 43
- Mouse operations 41
- Mouse wheel 140
- Moving the cursor 90, 96, 141
- MPE 1, 3, 11, 20, 24, 32, 53–61, 80, 91, 210–23, 211–13, 215–17, 222–23, 227–28, 232
 - compiling 183
 - environment 171, 173
 - notation 70
 - version 4.0 70
- MPE file labels 191
- MRU 77
 - search request 142
- Multi-line host commands 170
- MySQL
 - syntax coloring 123

N

- Native Language Support 56
- Network 53
 - install 7, 8
- Network requirements 2
- New features 4
- New Host File dialog 29, 59, 63
- Newline missing 81
- NLS 56
- Nonprinting characters 82, 91, 92
 - host command 168
- Notation
 - MPE 70
 - POSIX 70
 - Unix 69

O

- Open
 - application window 51
 - desktop icon 50
 - drag-and-drop 50, 51
 - from command line 75
 - host file 59, 79, 215
 - maximum length 81
 - selection 95
 - taskbar button 51

- unnumbered 81
- Open a copy 81
- Open dialog 46, 48, 64, 77
- Open Host File dialog 29, 80
- Oracle
 - syntax coloring 123
- Overwrite mode 90, 91

P

- Page setup 155
- Page size 157
- Parentheses
 - matching 122
- Password 56
 - dynamic 215, 219
 - prompting 59
- Paste Progress dialog 97
- PASXL command 183
- Pattern matching 143
- PID 212
- PIN 212
- ping 215, 218, 220
- Port 11, 216
 - MPE 1, 193
 - UNIX 1, 193, 218, 221
- Port number 19
- Position
 - Exchange 141
 - Go to 141
 - Save 141
- POSIX 70, 211
 - notation 70
- POSIX prefix 74
- Preferences dialog 13, 162, 232
- Print
 - host 158
 - host line numbers 156
 - line numbers 156
 - lines per page 157
 - manuals 11
 - margins 157
 - page headings 156
 - page setup 155
 - wrap lines 156
- Printer
 - host 159
- Process ID 212
- Proxy server 193
- Punctuation matching 122
 - enabling 122
 - unmatch alert 122

Q

- qedconn.dat 11, 12
- qederr.log 12
- Qedit workfile 188
- Qedit workfile 28, 33, 188
- Qeditmgr
 - MPE 20, 213, 227
 - UNIX 20, 214
- Qedservmode 20, 21
- QEDSERVMODE
 - host command 172
- QSL
 - file extension 50
 - registering 50
- qsl.doc.pdf 11
- qstart.pdf 11
- qwin.ini 11, 12
- qwin.pdf 11
- qwinddb.bkp 12
- qwinddb.dat 11, 12
- qwinerr.log 11
- qwintips.txt 11

R

- Read only 78, 82–88, 231
- Record length 64, 66
- Record Length
 - exceeded 82
- Record number 131, 146
- Recovery
 - HP-UX files 224
 - MPE file 223
- Rectangular selection 44
 - copying 105
 - cutting 105
 - deleting 106
 - extending 96, 106
 - inserting 105
 - pasting 105
- Redo 30, 108
- Registering file types 49
- Regular expression 143, 251
 - anchor 252
 - backreferences 256
 - character class 253
 - escape character 255
 - escaped sequences 255
 - metacharacter 251
- Remove scripts 177
- Rename
 - text file 153
 - workfile 162
- Rename workfile 162

- Replace 147
 - pattern matching 147
- Replace dialog 147
- Requirements
 - hardware 1
 - network 2
 - Software 1
- Revert 108
- Right-click 46, 47
- Roman8-1999 56
- Roman9 56
- RPM 1, 215–17, 222, 233
- Ruler 113, 124

S

- Save
 - all files 28, 164
 - as regular file 162
 - as workfile 154
 - automatic 166
 - position 141
- Save As dialog 163
- Script control 178
- Script Control dialog 178
- Script directories 176
- Script-based features 175
- Scripting language 175
- Scroll 140
 - bar 41, 95
 - box 225
 - mouse wheel 140
- Scroll lock key 90
- Security/3000 57, 215, 219
- Select All text 98, 153, 154
- Selecting
 - line 43
 - rectangle 44
 - word 42, 66
- Selection as host command 170
- Seleq 73
- SendTo 49
- Serial number 8–9
- Server
 - configuration 19, 159, 227
 - name 216, 219
 - trace 11
 - warning 82
- Server configuration
 - host command 172
- Server install 9
- Server name 19
- Server port number 19
- Server settings 213
- Server trace 208

- Shift column 102
- Shift text 103
- Shift with tab 103
- Shortcut 205
 - editing 205
 - menu 46, 189
 - searching 204
- Shortcut keys 204
- Smart search 64, 66
- Sort
 - columns 179
 - text 179
- Spaces
 - for tabs 67
 - trailing 68, 132
- Special characters 51, 91, 92, 189
- Split
 - box 41
 - collapsing 37
 - command 37
 - views 35, 124
- SQL
 - syntax coloring 123
- Square brackets
 - matching 122
- Status bar 30, 33, 40, 89
 - indicators 35
- Support files 11
- Switching applications 39
- Syntax coloring 122
 - Allbase 123
 - ambiguity 123
 - extension 123
 - language code 123
 - MySQL 123
 - Oracle 123
 - precedence 124
 - SQL 123

T

- Tab
 - character 67, 92
 - compress 104
 - expand 104
 - shift with 103
 - stop 114, 115, 214
- Tag 133, 134, 214
- TCP/IP 2, 53, 161, 193
- telnet 218, 221, 227
- Temporary connection 54
- Terminal emulator 64
- Terminator
 - CR 68
 - LF 68

- line 68
- Terminator, default line 137
- Text 91
 - adding 91
 - copying 98
 - deleting 100
 - finding 142
 - moving 99
 - pasting 98
 - replacing 146
 - selecting 43, 94
 - shift 103
 - workspace 33
- Tile
 - horizontal 40, 186
 - vertical 40, 186
- Time zone 226
- Timeout 194
- Tip of the day 22
- Tip of the Day dialog 22
- Title bar 24, 32, 39
- Toggle keys 31
- Toolbar 26, 45
 - buttons 27
 - moving 26
 - rectangular 27
- Trace
 - server 11
- Trace log file 217, 218, 220
- TRANCOMP command 183
- Transmission information 213
- Trial 216–17, 216–17
- Troubleshooting 207
- TXT
 - file extension 50
- TZ 226, 227, 228

U

- UDC, logon 20
- Undo 30, 108
 - transaction 107
- Undo all 108
- UNIX 1, 55, 57
 - host 53
 - notation 69
- Unix files 68
- Unload scripts 177
- Unnumbered file 81
- Upgrade 12
- User Defined Command, logon 20
- User labels, MPE 191

V

VESOFT 21, 215, 219

W

warning

folded lines 82

line number 83

server 82

Web site 4

Whole words 64, 66

Wildcards

directory 72

Open Host File 81

Win32S subsystem 1

Window

application 23

attributes 23

close 40, 164

control menu 32

document 31

maximize 40

minimize 40

restore 40

size 39

terminal 3, 188

Windows character set 56

Word search 143

Workfile 28, 33, 188, 232

rename 162

Save As 162