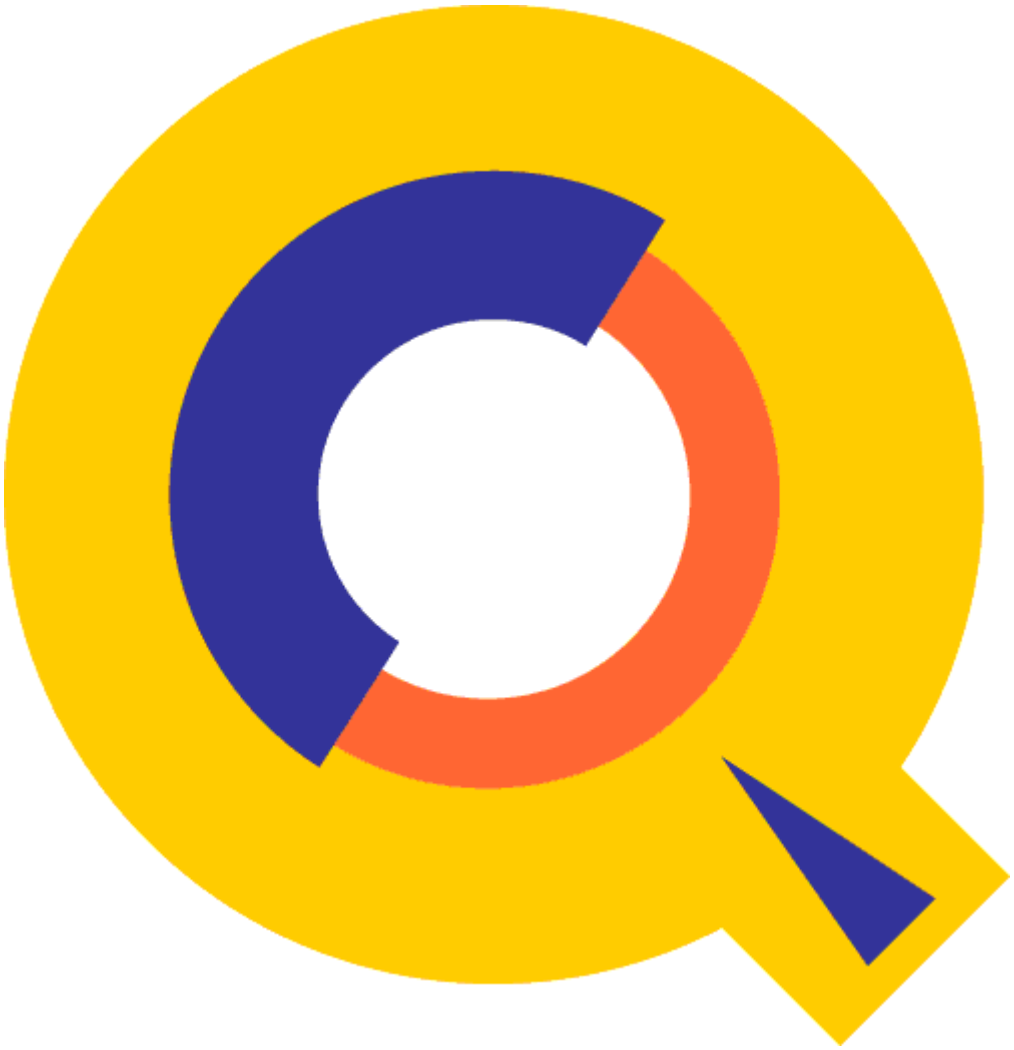
Qedit 6.7 for HP-UX

Change Notice

by Robelle Solutions Technology Inc.



Program and manual copyright © 1977-2026 Robelle Solutions Technology Inc.

Permission is granted to reprint this document (but not for profit), provided that copyright notice is given.

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

Robelle Solutions Technology Inc.

Suite 325-4025 Dorchester Road

Niagara Falls, ON, Canada L2E 7K8

Phone: 289.480.1060

E-mail:sales@robelle.com

E-mail:support@robelle.com

Web: www.robelle.com

Contents

Introducing Qedit/UX 6.7 1

Introduction 1
Qedit for Windows 1
Highlights In Version 6.7 2
Highlights In Version 6.6 2
Highlights In Version 6.5 2
Highlights In Version 6.4 2
Highlights In Version 6.3 2
Highlights In Version 6.2 3
Highlights In Version 6.1 3
Highlights In Version 6.0 3
Compatibility 3
CPU Serial Number (uname) 4
Known Problems 4
Documentation 4
User Manual 4
Change Notice 4
Printed Documentation 5
Web Download 5

Installation 7

Introduction 7
Installation Assistance 7
Step 1: Log On as Root 7
Step 2: Create Robelle Directory 8
Step 3: Restore Files 8
Step 4: Setting Up Variables 8
Step 5: Setting Up Links 9
Step 6: Start the Qedit for Windows Server (optional) 10

Enhancements in 6.5 11

Introduction 11

Enhancements in 6.4 12

Introduction 12

Enhancements in 6.3 13

Introduction 13
Set Keep Var Off LF on 13

Qedit for Windows 15

Introduction 15

Server Process 15
HP-UX 10.20/11.0 15
Port Number 17

Bugs Fixed **19**

Bugs Fixed Since Version 6.1 19

Introducing Qedit/UX 6.7

Introduction

Qedit is our fast and efficient full-screen editor and development home base for HP-UX systems, both workstations and minicomputers. One of our objectives for Qedit is to give PowerHouse and COBOL developers all the necessary capabilities to perform their work from a single program.

We are constantly enhancing Qedit with new features and fixes requested by users. Not only does this change notice include a description of Qedit's enhancements, it includes detailed installation steps and a list of other changes for host-based Qedit. This change notice also includes a description of the server portion of Qedit for Windows.

Host-based Qedit is also available for MPE, with virtually identical features. If you would like more information, please contact Robelle Solutions Technology Inc.

Qedit for Windows

Our newest editor, Qedit for Windows, now gives you the familiar Windows interface for editing both MPE/iX and HP-UX files, all from a single program. Qedit for Windows is the next generation of our popular editor, which consists of a Windows editing client and either an MPE/iX or HP-UX editing server that work together to edit host files. Since its release, we have focused our development efforts on Qedit for Windows, adding more functionality and continuing to expand its capabilities. If you would like to upgrade to Qedit for Windows, please contact Robelle Solutions Technology Inc.

Highlights In Version 6.7

This is an overview of all the changes implemented in this version.

- Qedit 6.7 has been tested with Jump/3000 the 2028 mitigation software.

Highlights In Version 6.6

This is an overview of all the changes implemented in this version.

- Visual Mode for HP terminals on HP-UX, will automatically turn on set Vis Blockemulation on. (HP-UX only)
- Visual mode for Qedit/Open will no longer fail with unrecoverable file system error. (Linux Only)

Highlights In Version 6.5

This is an overview of all the changes implemented in this version.

- Visual Mode for HP terminals has been added to Qedit/Open. (Linux Only)
- The behaviour of regexp window has been fixed in one corner case. (All platforms)
- Help files are now working in Qedit/Open. (Linux Only)
- Hold files are now working in visual and in other areas of Qedit/Open. (Linux Only)
- An issue was fixed when Qedit repainted the updated screen incorrectly after a block copy or move is now fixed.
- Set vis above can only be 0. This is temporary. (Linux Only)
- Set Vis BlockEmulation has been turned on by default. (Linux Only)

Highlights In Version 6.4

- Qedit has been ported to Linux, and is available with Line mode features.

Highlights In Version 6.3

- Qedit for HP-UX now writes out the size of a fixed length file when making a file fixed length, with set keep var off. It will also write out a file as to whether or not Line Feeds are on or

off. The files are local to your working directory under the names .qxrecsize and .qxlf.

- The Qedit Server now supports login via PAM and which supports LDAP.

Highlights In Version 6.2

- The Qedit Server would not properly honour an adminstrative lock on a user account login.

Highlights In Version 6.1

- The Verify command incorrectly parsed certain entries near the end of the verb table such as ZZ, String etc.

Highlights In Version 6.0

- Qedit now recognizes files with the extension of .COB and .cob as Cobol files on HP-UX.
- Qedit on HP-UX now has the command set compatvarsub on which does not replace $-variables that are not defined with spaces. It simply passes the $token, thru to the Qedit command interpreter if the variable is not defined.

- Qedit now recognizes files with the extension of .COB and .cob as Cobol files on HP-UX.
- Qedit now support long user names on HP-UX 11.31, implemented by lugadmin.
- Qedit now supports the HP-UX Long Password feature on HP-UX 11.31. Error messages have been improved for login issues.
- Qedit would return a status code of 123 upon exit even if an error had not occurred.

Compatibility

Qedit/UX is compatible with most versions of HP-UX 10.20 and later.

IMPORTANT: As of HP-UX 11.0, HP has dropped support for block-mode terminals. For this reason, full-screen editing is available in Screen mode or by enabling **Set Visual Blockemulation ON** option on HP-UX 11.0 and later.

Qedit commands are added to the Redo stack before variable substitution occurs (**Set Varsub On**) i.e., with the variable name. So, if the variable value changes between the time the command is entered and the time it is retrieved from the stack, the results may be different. It's also important to note that commands related to Redo stack operations such as **Listredo**, **Do**, **Before** cannot have trailing comments enclosed in curly braces anymore. The comments are not removed and likely cause a syntax error.

In the examples in this document, we have used HP-UX 10.20 conventions and assumed Qedit is installed in the /opt/robelle directory.

CPU Serial Number (uname)

Qedit/UX runs only on CPUs whose serial numbers (the "uname" on HP-UX) have been encoded. If it fails to run and you see an "invalid HPSUSAN" error message, call Robelle for assistance.

Known Problems

There are no known problems at the moment.

Documentation

Qedit comes with a User Manual and a Change Notice. You may have received printed copies of these.

They are also available as PDF or HTML files. You can download the files from the Robelle web site.

User Manual

The user manual contains the full description of all the Qedit commands, as well as usage tips. The manual is up-to-date with all the latest changes incorporated in Qedit. To see only the changes in the latest version, see the "What's New" section of the manual, or see the change notice.

Change Notice

For a complete description of the latest changes made to Qedit, the installation instructions, and any compatibility issues, see the change notice that was included with the release.

Printed Documentation

The latest user manual and change notice are available in Adobe PDF format. If you do not already have the Adobe Acrobat Reader, you can get a copy from http://www.adobe.com/prodindex/acrobat/readstep.html.

Web Download

You can download the files from the Robelle web site at http://www.robelle.com/library/manuals/.

Installation

Introduction

If you are a *new Qedit/UX user*, please follow the installation instructions in the *Qedit User Manual*. To install this upgrade of Qedit, follow these steps:

1. You must log on as root.
2. Create the correct directory structure.
3. Restore Qedit/UX and its associated files from the distribution tape.
4. Set up variables to let Qedit know where some of its supporting files are located.
5. Set up a link between /var/tmp and /usr/tmp if you are installing on HP-UX 10..*x*.
6. If you have the Qedit for Windows server, you need to start the daemon process. (optional)

The installation should take about half an hour, during which time no one can use Qedit.

Installation Assistance

If you have any questions about the upgrade or run into any problems, please call us. Technical support is available on weekdays from 7 a.m. to 4 p.m., Pacific time.

Step 1: Log On as Root

There are two ways you can log on as root:

1. Exit from HP-UX and log on with root as the user name.

2. If you are already logged on, you can execute this command:
   ```
   su -
   ```

In either case, you have to know and supply the user password for root.

Step 2: Create Robelle Directory

Before restoring files, you must first create the directory in which Qedit will reside:

```
mkdir /opt/robelle
```

Step 3: Restore Files

Use the following command to restore the Qedit/UX files from the distribution tape:

```
tar xv /opt/robelle
```

This command assumes your tape device is /dev/rmt/0m. If it is not, you need to specify your tape device using the "f" option in the tar command. For example, if your tape device is /dev/rmt/1m, you need to use the following command to restore the files:

```
tar xvf /dev/rmt/1m /opt/robelle
```

Step 4: Setting Up Variables

You can set up numerous variables to make running any of the Robelle products easier.

The three variables that we recommend setting are ROBELLE, PATH and MANPATH.

- The ROBELLE variable tells the Robelle products where it's supporting files are located.
- The PATH variable, when set, allows the user to run Qedit by just typing "qedit" instead of the fully qualified program name.
- The MANPATH variable, when set, allows users to access the manpage (Help page) for Qedit.

You can invoke Qedit with the following command:

```
/opt/robelle/bin/qedit
```

The easiest way to add variables accessible to all users is to set the variables in the global exec file for the shell you use.

For the Bourne and Korn Shells we recommend the following additions:

```
export ROBELLE=/opt/robelle
export PATH=$PATH: $ROBELLE/bin
export MANPATH=$MANPATH: $ROBELLE/man
```

Remember to delete any PATH or MANPATH settings in /etc/d.profile, so that new users do not override your changes. You also have to warn existing Bourne and Korn shell users to change their .profile file in their home directories.

For the C Shell we recommend that you add the following commands after any existing PATH or MANPATH statements in the /etc/csh.login file:

```
setenv ROBELLE /opt/robelle
set path=($path $ROBELLE/bin)
setenv MANPATH "$MANPATH": $ROBELLE/man
```

Remember to delete any PATH or MANPATH settings in both /etc/d.login and /etc/d.schrc, so that new users do not override your changes. You also have to warn existing C shell users to change their .login and .cshrc files in their home directories.

Step 5: Setting Up Links

When you run Qedit, it creates and uses certain temporary files. On HP-UX 10.*x* systems, temporary files are generally created in the /var/tmp directory. To get Qedit to keep an undo log and redo stack, you must set up a link between /usr/tmp and /var/tmp. To check whether this has already been done, enter

```
ls -l /usr/tmp
```

If a link exists, you will see a line such as this:

```
lrwxrwxr-t 1 root sys    8 Aug 16  1996
/usr/tmp -> /var/tmp
```

If the link is not set up, you can create it by entering

```
ln -s /var/tmp /usr/tmp
```

Step 6: Start the Qedit for Windows Server (optional)

If you have the Qedit for Windows server software, you must start the Qedit/UX daemon process before any Qedit clients can connect to your HP-UX machine. To allow users to connect to the Qedit/UX daemon process, you must log on as root and issue this command:

```
qedit -d
```

The Qedit server process requires three log files. By default, these files are located in the following directory:

```
/opt/robelle/log/qedit/
```

If you have moved Qedit to a different directory, you will have to perform the following steps:

1. Set the ROBELLE environment variable with the new directory name before you start the Qedit server process.
2. Make sure the new directory has exactly the same structure as the /opt/robelle directory.

Enhancements in 6.5

Introduction

The following sections describe the new enhancements to Qedit/UX since the last major release. The number in brackets at the end of each section indicates the version in which the feature was first introduced.

- Qedit has been ported to Linux, specifically the Line mode version and all the associated commands have been ported and tested. Next step will be visual, screen and server modes.

Enhancements in 6.4

Introduction

The following sections describe the new enhancements to Qedit/UX since the last major release. The number in brackets at the end of each section indicates the version in which the feature was first introduced.

- Qedit has been ported to Linux, specifically the Line mode version and all the associated commands have been ported and tested. Next step will be visual, screen and server modes.

Enhancements in 6.3

Introduction

The following sections describe the new enhancements to Qedit/UX since the last major release. The number in brackets at the end of each section indicates the version in which the feature was first introduced.

- The Qedit server supports login via PAM and in turn LDAP/UX.
- Qedit for HP-UX now writes out the size of a fixed length file when making a file fixed length, with set keep var off. It will also write out a file as to whether or not Line Feeds are on or off. The files are local to your working directory under the names .qxrecsize and .qxlf. See the Keep command for details under the section Variables for Suprtool.

Set Keep Var Off LF on

Their is often a need to import variable length CSV files into a format that can be easily imported into Suprtool on HP-UX. In order for Suprtool to input a file that is not self-describing, we need to make a variable length file fixed length, and Suprtool needs to know the record size and whether or not the file has line feeds or not.

Qedit 6.3 will now create two files if you have set that the file will be fixed and the Line Feeds are on. This tells Qedit to write out the record size of the file, which we can then export by catting the file, and similarly we can do the same with whether or not the file has linefeeds, so the communication between Qedit and Suprtool is almost automatic. We use the method of using cat, so that the variable can be set outside of qedit such that the suprtool process will be able to reference the variable. Currently these two files are only created if set keep var is of and lf is on.

```
qedit << !EOD
t newfile,unn
set keep var off lf on
keep fixedfile,unn
exit
!EOD
export ROBQXRECSIZE=`cat .qxrecsize`
export ROBQXLF=`cat .qxlf`

suprtool << !EOD

set varsub on
input myfile, rec $ROBQXRECSIZE, LF
```

Qedit for Windows

Introduction

Qedit for Windows lets you edit local MPE/iX and HP-UX files from a single MS Windows program. It consists of a Windows editing client and either an MPE/iX or HP-UX editing server that work together to edit your host files. To take advantage of Qedit for Windows, you need both the Qedit client and the Qedit server.

Qedit for Windows uses the popular TCP/IP protocol for communicating between the client and the server (this is the same protocol that you use to access the Web). Configuring the Qedit server software requires creating the correct TCP/IP environment for Qedit for Windows.

Server Process

Qedit clients can connect to Qedit/UX only if the Qedit/UX daemon process is running (the Qedit/UX daemon process cannot be started from inetd). This process listens for connections on a registered port number (described below). To allow users to connect to the Qedit/UX daemon process, you must log on as root and issue this command:

```
qedit -d
```

The Qedit/UX daemon process should always be running, so it is a good idea to automatically start the daemon as part of the system startup process. On HP-UX 9.0, this is done by adding the following command to the file /etc/rc:

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

HP-UX 10.20/11.0

On HP-UX 10.20 and up, you do not modify the startup shell script. Instead, you need to create a number of files. The HP-UX documentation states that the following characters cannot be used as

part of the file names: `[.,~#]`. Otherwise, you can choose any name for these files, as long as the names are consistent throughout the process. In our example, we use qedit_server.

You first need to create a control file in /etc/rc.config.d. This file sets a control variable that will be checked by the startup script. If the control variable is set to 1, the server will start; if it is not equal to 1, the server will not start. We will use QEDIT_SERVER as our variable name. The /etc/rc.config.d/qedit_server control file will now contain the following:

```
# ****** File:  /etc/rc.config.d/qedit_server
******
# Qedit for Windows server configuration.
#
# QEDIT_SERVER:    Set to 1 to start
#                  Qedit for Windows server

QEDIT_SERVER=1
```

Next, you need a shell script that will actually start the server. You can copy a template of the shell script from /sbin/init.d/template.

```
cd /sbin/init.d
cp template qedit_server
```

Modify the file so that it contains the necessary commands to start the server. You have to change all occurrences of CONTROL_VARIABLE to the variable name you used in the control file (i.e., QEDIT_SERVER).

You also need the execute command for the server program. Insert this command (underlined below) in the section after the 'start') string. The section looks like this:

```
'start')

      # source the system configuration
variables
      if [ -f /etc/rc.config ] ; then
               . /etc/rc.config
      else
         echo "ERROR: /etc/rc.config defaults
file MISSING"
      fi

      # Check to see if this script is allowed
to run...
      if [ "$QEDIT_SERVER" != 1 ]; then
         rval=2
      else

      # Execute the commands to start your
subsystem
         /opt/robelle/bin/qedit -d
      fi
      ;;
```

Finally, you need a symbolic link to specify when the script in /sbin/init.d will be executed at boot time. Typically, you would start the server as the last step at run level 3. Get a list of all the startup files in /sbin/rc3.d with

```
ls /sbin/rc3.d/S*
```

Link names in this directory follow a set of conventions. The names start with the letter *S* or *K*. S links are startup scripts; K links are shutdown or "kill" scripts. The next three characters in the name represent an execution sequence number. This number must be 3 digits, and its value should be a number greater than the highest value on the ls listing. For example, if the last link is called S100nfs.server, you could use S111qedit_server. Create the symbolic link with

```
ln -s /sbin/init.d/qedit_server
/sbin/rc3.d/S111qedit_server
```

For the time being, you do not need a "kill" link.

Port Number

By default, Qedit/UX listens on port number 7395. This port number has been registered with the Internet Naming Authority, so you should not have any conflicts with other HP-UX tools for the same port

number. If there is a conflict, you can start the Qedit server process with a different port number. For example,

```
qedit -d5678
```

tells Qedit to listen to port number 5678 instead of the default number (7395). If you change the port number on the Qedit/UX server, you must also change the port number on every Qedit client to the same value (5678 in this example). Client port numbers can be changed in the Server Settings dialog box of the Options menu.

If you want listings from netstat and other networking tools to identify the port number as "qwin" instead of just "7395," you have to change the /etc/services file so that it includes the Qedit port number.

```
qwin    7395/tcp     #Robelle Qedit for Windows
```

Bugs Fixed

Bugs Fixed Since Version 6.1

The following problems have been fixed since the last production version.

- The Qedit server would not properly honour an administrative lock on a user account, during login from the Qedit for Windows client. [6.1.01]