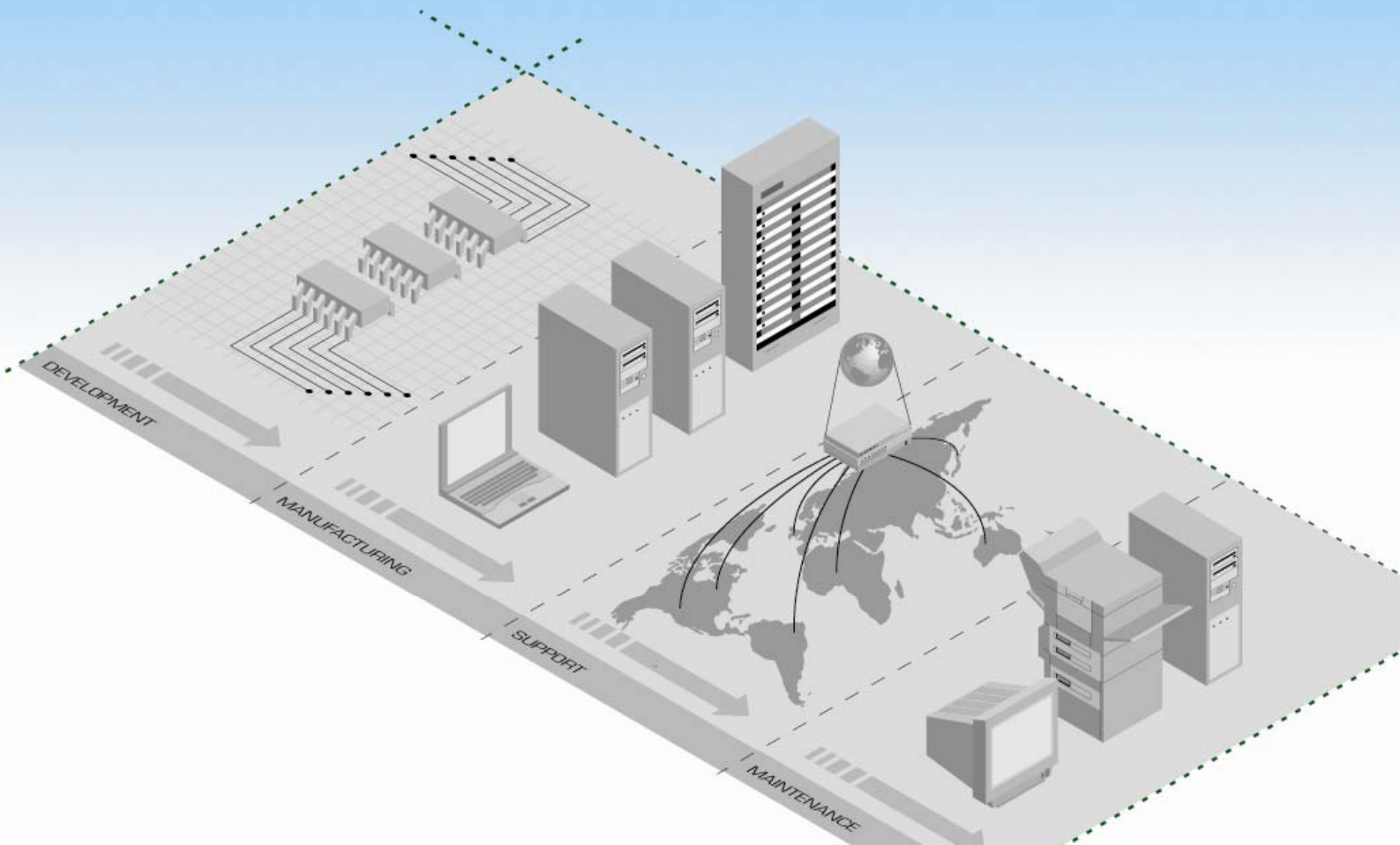**Eurosoft**
pc reliability solutions

# QA+WIN32 ™
## PC Reliability Solutions

## User Manual

*Nobody checks out PCs better than Eurosoft* [SM]

Eurosoft
PC Reliability
Solutions

# QA+WIN32™

# User Guide and
# Reference Manual

Eurosoft (UK) Ltd
3 St Stephen's Road
Bournemouth
Dorset BH2 6JL
England

Tel: +44 (0)1202 297315   Fax: +44 (0)1202 558280
EMAIL: info@eurosoft-uk.com
Web Site: http://www.eurosoft-uk.com

102004MVQAPV8.33

# CONTACT DETAILS

## Eurosoft (UK) Ltd.

Head Office
3 St. Stephen's Road
Bournemouth
Dorset BH2 6JL
United Kingdom
Tel +44 (0)1202 297315
Fax +44 (0)1202 558280
Email: info@eurosoft-uk.com
Website: www.eurosoft-uk.com

## Eurosoft (US) Inc.

North American Sales Office
Suite 101, One Davol Square
Providence, RI 02903
USA
US Toll Free Tel +1 (888) 980-9595
International Tel +1 (401) 751-9595
Fax +1 (401) 751-0088
Email: info@eurosoft-us.com
Website: www.eurosoft-us.com

## Information & Sales

| | |
|---|---|
| Europe | info@eurosoft-uk.com |
| United States | info@eurosoft-us.com |
| Pacific Rim | info@eurosoft-au.com |

## Support

| | |
|---|---|
| Europe | support@eurosoft-uk.com |
| United States | support@eurosoft-us.com |
| Pacific Rim | support@eurosoft-au.com |

# Technical Support

If you have any problems getting QA+WIN32 to run properly, contact your nearest Eurosoft office by e-mail, telephone, fax, or letter. The contact details are:

## Eurosoft (UK) Ltd.

### Head Office

3 St. Stephen's Road
Bournemouth
Dorset BH2 6JL
United Kingdom

Tel  +44 (0)1202 297315

Fax +44 (0)1202 558280

Email: support@eurosoft-uk.com

Website: www.eurosoft-uk.com

## Eurosoft (US) Inc.

### Support Office

310 Third Avenue NE
Suite 108
Issaquah, WA 98027-3346
USA

Tel  +1 (425) 369-8582

Fax +1 (425) 642-8088

Email: support@eurosoft-us.com

Website: www.eurosoft-us.com

## Support

| | |
|---|---|
| Europe | support@eurosoft-uk.com |
| United States | support@eurosoft-us.com |
| Pacific Rim | support@eurosoft-au.com |

This page is intentionally left blank

# Table of Contents

This page is intentionally left blank

# Chapter 1  Before You Begin

*Before You Begin*

*Inside this chapter …*

- ■  About this Book
- ■  System Requirements

QA+WIN32 is the ultimate in Windows® PC diagnostic software. QA+WIN32 provides the flexibility technical service personnel require. QA+WIN32 provides an extensive set of advanced diagnostic functions for end users, service and support professionals, and factory technicians to troubleshoot even the most difficult PC problems. The QA+WIN32 diagnostic application has the following key features to help you in your troubleshooting efforts:

■ **Advanced Diagnostic Tests**

QA+WIN32 diagnostic tests reliably identify most problems associated with the major components of PC compatible systems. It tests the system board CPU, NPU and chipsets, hard and floppy drives, CD-ROM drives, RAM, serial and parallel ports, video adapter, and identifies PCMCIA sockets and cards, Plug and Play version and nodes, printer, keyboard and mouse.

■ **Flexible Test Control**

The extensive test options with the QA+WIN32 Windows® based diagnostics let you run groups of tests in a batch, specify parameters to use for each test (e.g., video modes, disk cylinders, etc.), specify the number of passes you wish to run, specify the order in which the tests will run (tests can also be repeated), log the test results to a log file, and save test settings as a script for future use.

■ **Detailed System Information**

The SYS group tests that are part of QA+WIN32 provide detailed information on PC configuration, IRQ/DMA assignments, memory usage, device drivers and more.

## About this Book

This book will help you install and use the QA+WIN32 diagnostic software tools. It contains complete information about the diagnostic test capabilities of QA+WIN32 for the end user and the factory technician.

This book uses the following text formats and symbols:

| Convention | Description |
|---|---|
| **Boldface** | Indicates actual QA+WIN32 screen objects, such as menu items that you select or click. |
| ⌇🖱 | Indicates a procedure. |
| ■ | Indicates a list of related information. |
| [CTRL] | Indicates a specific key that you press. |
| [CTRL] + [ALT] | Indicates a specific key combination that you press. |
| Courier | Indicates user input in procedures, email addresses, and web URLs. |

## System Requirements

- ■ Pentium, Pentium II, Pentium III, Pentium IV or higher compatible computer
- ■ Sufficient memory to run Windows® 9x or higher.
- ■ ISA, VESA Local Bus, PCMCIA or PCI system bus.
- ■ CDROM/DVD Disk Drive
- ■ Windows® 9x (or higher) operating system.

This page is intentionally blank

# Chapter 2  QA+WIN32

## *QA+WIN32*

*Inside this chapter …*

- Starting QA+WIN32
- Changing Views
- Running Tests
- Printing Reports
- Creating Scripts
- Command Line Parameters

You can access the help system at any time from within QA+WIN32. You can also access help without running QA+WIN32. This can be accomplished by double-clicking qawin32.hlp located in *path*:\QAWIN32.

**To access the help system from within QA+WIN32, do the following:**

1.  Select **Help/Help Topics**. Help will display**.**

2.  Select the Help Topic you want to read.

    or

1.  Click the Help button, [?], on the top toolbar. Help will display.

2.   Scroll to the Help Topic you want to read.

**Note:** Help cannot be displayed on Windows PE.

## Starting QA+WIN32

After you launch QA+WIN32, the application appears in a full screen. At the top of the screen you see the QA+WIN32 menu. Under the menu are two toolbars. The first is the Utilities Toolbar. Underneath the Utilities Toolbar is the Group Toolbar.

The bottom of the screen is divided into two panes. The left half contains a listing of all groups, group components, and tests. You can change this view by

pressing the [+], [++], [+++] buttons or the ⊞ icon in the Tree View.

The right half of the screen is divided into six tabs: General, Test Information, Device Information, Test Settings, Test Results, and the Results Log. Click on a tab to change the view to that tab. You can also tile the tabs. To tile the view, choose **Tile View** from the **View** menu. This displays Test Results, Test Information, Test Settings and Results Log.

By default QA+Win32 starts in interactive mode, enabling tests that require user input to be run.  To change to non-interactive mode, deselect the View-Interactive menu option.  Note: tests that require user input will now skip.

**Note:** if you change to non-interactive mode and close QA+Win32, the next time you re-start the application it will remain in non-interactive mode.
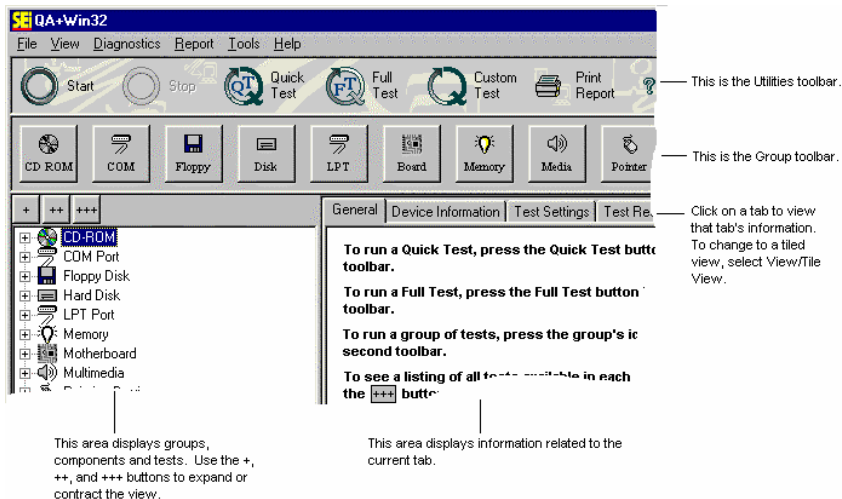
## Changing the Information View

When QA+WIN32 opens, six tabs are visible: General, Test Information, Device Information, Test Settings, Test Results, and Results Log

**To change the information view**, do the following:

1.  Select **View/Customize Information View**.

2.  Select Tile View or Tab View. Tab View displays the windows on top of each other, with one window visible at a time. Tile View displays the available tabs and the information contained in those tabs.

3.  Select the windows you want displayed.

**Note:** In Tile View, neither the General nor the Device Information tabs are available.

To switch windows in Tab View, click on the tab you want to view.

If you are in Tab View and want to display the windows in Tile View, select **View/Tile View** from the **View** menu.

To turn the status bar off, select **Status Bar** from the **View** menu.

**Note:** If you deselect any windows with the 'Customise Information View' dialog box then close QA+Win32, the next time you re-start the application these windows will still be deselected.  To view them again you must reselect them.

**Note:** The Test Information pane is blank on Windows PE (.rtf files cannot be displayed on PE)

## Other ways of changing the appearance of QA+WIN32

You may change the appearance of the interface by toggling the **Tile View**, **Group Toolbar**, **Tree View**, and **Status Bar** menu items in the **View** menu.

## Changing test options

There are 2 test options, Stop On Fail and Lap Count/Duration, both accessed by selecting **Options** from the **Tools** menu.  If Stop On Fail is enabled, QA+Win32 will stop testing when it encounters a test failure; by default this option is disabled.  The Lap Count/Duration setting allows you to specify how long testing should continue, either in terms of passes or duration in minutes; the default setting is a duration of 1 minute.

**Note:** If you specify a duration in minutes, testing won't stop immediately when the specified duration passes.  Instead, the currently running test or group will be allowed to complete first.

To turn on **Stop on Fail**, do the following:

1.  Select **Tools/Options**

2.  Check the **Stop Running When a Test Fails** box

3.  Click OK


To specify how long testing should continue for, do the following:

1.  Select **Tools/Options**

2.  Click on the down arrow

3.  Select **Lap Count** or **Duration**

Set the number of test runs or length of time for testing

**Note:** If you specify a lap count or duration and then close QA+Win32, the next time you re-start the application it remembers the value you set.


## Running a Quick Test

QA+WIN32 allows you to perform a quick check of your system. The Quick Test is a default set of tests designed to run quickly and test the majority of the hardware in your system. If your system lacks support for any of these tests, for instance if there is no audio card, then the line 'Group not available' is written to the file QckScrEr.Txt for each such test.

**Note:**  Be aware the Quick Test contains a potentially destructive IDE drive test.  The only way it would be destructive is if the system is restarted prior to the test completion or if the system locks during the Zone test for the IDE drive.  The Quick

Test also involves interactive testing that requires the user to actively participate by using the mouse and keyboard. **To run Quick Test**, do the following:

Select **Diagnostics/Quick Test**.

or

Click the **Quick Test** button, , on the toolbar

## Running a Full Test

QA+Win32 allows you to perform a full check of your system.  The Full Test is a default set of tests designed to test the majority of the hardware in your system.  For each device tested the entire test group is run.  If your system lacks support for any of these tests, for instance if there is no audio card, then the line 'Group not available' is written to the file FulScrEr.Txt for each such test.

**Note: Be aware the Full Test contains potentially destructive IDE drive tests and destructive floppy drive tests.**  The only way it would be destructive is if the system is restarted prior to the test completion or if the system locks during the tests.  The Full Test also involves interactive testing that requires the user to actively participate by using the mouse and keyboard.

**To run Full Test**, do the following:

Select **Diagnostics/Full Test.**

or

Click the **Full Test** button, , on the toolbar

## Testing a group

**To run a group of tests**, do the following:

Click the group's icon on the toolbar.

or

Double click on the name of the group in the Test List.

or

Highlight the name of the group in the Test List. Press the **Start** button, , on the Utilities toolbar.

## Testing one component

You may have more than one component within a group. Each group's components are listed beneath the group name. If only one component is listed, there is only one component available. For example, if you have one serial port, the COM group will list COM1 only.  If you have two serial ports, the COM group will list COM1 and COM2.

**To test one component**, do the following:

1.  Expand the Test List to the second level. You can do this by pressing the ⊞ next to the group name. Or press ++ . This will list each group's components.

2.  Double-click the component you wish to test.

**Note:** Tests may differ between components within the same group based on the tests supported for each device.

# Running one test

**To run one test**, do the following:

1. In the Test List, click the ⊞ next to the group. This will list the tests in that group.

2. Choose the test you wish to run.

3. Double-click that test.

    or

    Highlight the test and press the **Start** button, ⬤, on the toolbar.

# Running a combination of tests

You can select a variety of tests from the Test List to run as a series. This includes individual tests, components or entire groups.

**To select a combination of tests**, do the following:

1. Press the ⊞ button to expand the Test List.

2. Select the first test, component, or group you want to test.

3. Using the [CTRL] key, select each additional test, component, or group you want to test. To select a series of tests, components, or groups, use the [SHIFT] key.

4. When you are finished selecting tests, press the ⬤ Start button.

# Printing reports

From the **Report** menu, you can choose to display a Hardware Configuration Report or to print either a Full or Summary Report of your testing session.

The Hardware Configuration Report displays hardware configuration information in a dialog box.  This is the same information as is obtained by running the System Information group's Hardware Configuration test (test 205).

Of the printed reports, the Summary Report contains the following information:

- Product Information
- Test Summary – a list of the groups tested and whether the group passed or failed (it does not list individual tests run).  If a single test in a group fails, then the report displays the group status as 'FAILED'.

The Full Report comprises the Summary Report followed by a breakdown of the individual test results in each group.

**Note:** Neither the Summary nor Full Report can be printed from Windows PE.

## Results log file

The results of each test session are displayed on the Results Log tab and written to the log file, 'testing.log'. Each entry in the log file is preceded by a string of numeric fields, representing data such the date, time, group and test number, and test status. For instance, the following example interprets a typical entry's numeric fields:

04222003:1203: 300:207:1:2:3:1:Starting Word Length Test

- The first field is the *month day year* before the first colon. So this log was run on April 22nd 2003.
- The next field prior to the second colon is the time, which in this case is 12:03 PM. This is a 24-hour clock.
- The next field up to the third colon is the group number. In this case, the group number is 300, for COM.
- The next field up to the fourth colon is the test number. In this case, COM Test 207 – Word Length Test is being run.
- The next field up to the 5th colon is the LUN number. This indicates which item is testing, if you have multiple items, such as hard drives. In this above example, COM1 is being tested.
- The next field up to the 6th colon is the Type of Message. Here's a list of the potential values:

  2  =  Status
  3  =  Pass
  4  =  Skip
  5  =  Fail
  6  =  Abort
  13 =  Report
  14 =  User query

So in the above example, the Type is "Status".

In this example:

04222003:1203: 300:207:1:**3**:3:1:Test Passed

The Type is "Pass".

Whereas, in this example:

04222003:1203: 300:205:2:**5**:3:1:Test failed

The Type is "Fail".

There are additional "Types" but they are obsolete and no longer used.

- The next field up to the 7th colon is the Action message. There are three options:

  1  =  Start
  2  =  Append
  3  =  End

However, this field is typically 3. This is the "End" action, which outputs the results of internal processing to the log.

- The last field just before the text is the "Lap" number. So if you run multiple laps, that information will be output into that field.

- Finally, we have the text description which is "Starting Word Length Test".

**Note:** If the system under test doesn't support all the groups or tests listed in the 3 pre-written script files (QAWin32Q.dat, QAWin32F.dat and QAWin32C.dat), then 3 error

lines will be written to the top of the script.  These explain that the unsupported groups/tests have been recorded in the respective script error files (QckScrEr.Txt, FulScrEr.Txt and CstScrEr.Txt).

## Changing test settings

The Test Settings tab displays tunable parameters for a variety of tests. You can change the parameters of a test if you wish.

**To change the test settings**, do the following:

1. Double-click the setting you want to change. A cursor will appear.

2. Make whatever changes you want to make to the setting.

3. Click outside of the test setting.

## Customizing a test script

QA+WIN32 allows you to select and save a sequence of tests. The saved test sequence is called a script. If you know that you always want to run a certain selection of tests, you should set up a customized test sequence and save the sequence as a script. You can also write a test script using any text editor, or Eurosoft's script writing utility, VQAScript.

**To customize a test sequence**, do the following:

1. Select **Tools/Customize Test…**

2. Using the icons, select the tests you want to run. Tests marked with a ✓ are tests selected. Click on a test to select or deselect it. You can check the group name or component name to select an entire group or component.

   **Note:** If you want to run most, but not all tests, click the **Select All** button. Then deselect the tests you do not want. This is useful if you are running most, but not all, tests.  Likewise, if you only want to run a few tests, click the **Deselect All** button to clear the test list.  Then select only the tests you want to run.

   **Note:** By default the tests of the Custom Test script, QAWin32C.dat, are selected.

3. Change any test settings. Highlight the test you want to change. Right mouse click and select **Properties**. Select the **Test Settings** tab. Double click a setting to make changes to it.

4. Click **OK** to accept setting changes. Click **Cancel** to close the dialog box without accepting changes.

5. Click **OK** to save the custom sequence. Click **Cancel** to close the dialog box without saving the sequence.

6. From the menu bar, select **File/Save Custom Test Script As**. The **Save As** dialog box opens. Navigate to the QAWIN32 directory, or other directory where you want the script located. In **File name**, enter the name of the script. If you want to overwrite the current custom script, select **File/Save Custom Test Script.**

7.  Click **OK** to save the script. Click **Cancel** to close the dialog box without saving.

**Note:** If you do not save the current session as a script, you will not save the custom test information. The custom test setup is lost once you exit QA+WIN32 unless you save it.

## Writing a test script

You can write a custom test script using any text editor. The file must follow certain syntax and must be saved as a .dat file. Examples are shown below.

The available syntax is:

| | | |
|---|---|---|
| <Start> | :== | <sentence> |
| <sentence> | :== | { <select> \| <unselect> \| <grp_param> \| <tst_param> } |
| <select> | :== | "select" <grpid> <lun> <test#> |
| <unselect> | :== | "unselect" <grpid> <lun> <test#> |
| <grp_param> | :== | "group" "param" <grpid> <lun> {<title> \| <tag> } "=" <data> |
| <tst_param> | :== | "test" "param" <grpid> <lun> <test#> {<title> \| <tag> } "=" <data> |
| <title> | :== | <string> |
| <data> | :== | <string> \| <integer> \| <bool> \| <metric> \| <pctg> |
| <bool> | :== | "true" \| "false" |
| <metric> | :== | "min" \| "mid" \| "max" |
| <pctg> | :== | "percent" "(" <integer> "," <metric> ")" |

Specific semantics:

**<grpid>**   unsigned integer representing a group that was loaded by a UI.

**<integer>**   signed/unsigned sequence of digits, may be preceded by 0x or 0X to denote a hex value.

**<lun>**   unsigned integer representing a lun associated with a loaded group. Luns normally range from 1 to N, but a lun value of 0 implies that the operation applies to all luns in the group.   A lun is a logical unit number and represents each device in the group on the machine, i.e. if you have two COM ports you would have Lun 1 (representing COM1) and Lun 2 (representing COM2).

**<string>**   any ASCII data delimited by '"'. Use the standard C escape sequence to include a delimiter within a string. No other C escape sequences are supported, although strings may be continued with \ <EOL> (as in ANSI C).

**<tag>**   unsigned integer representing a parameter tag ID, which uniquely identifies a group or test parameter. Tag IDs are created by diagnostic DLLs.

**<test#>**    unsigned integer representing a valid test # for a group loaded by a UI. The requested action will be applied only to tests associated with the specified lun, unless a lun of 0 is specified. Test #s normally range from 201 to N, but a test # of 0 implies that the operation applies to all tests for the specified lun.

**<title>**    string that uniquely identifies a group or test parameter. Titles are specified by the designer of a diagnostic DLL.

General semantics:

- A <title> is treated as a case insensitive object (i.e., case insensitive comparisons will be used when scanning for a match against API I97 diagnostic data).

- Assigned <data> is rejected if it exceeds a parameter's max and min bounds.

- Assigned <data> is rejected if its type is incompatible with the target parameter's type (e.g., assigning <string> to a numeric parameter, assigning <bool> to a numeric parameter, assigning <metric> to a <bool> or <string>, etc.)

- Metric values are extracted directly from parameter information supplied by diagnostic designers, i.e., each numeric test parameter has a default value, as well as a minimum and maximum value. The "mid" metric is computed using this formula:

(parameter "min" + parameter "max") / 2

- The <pctg> value is computed as a simple, integer percentage of the parameter's min, mid, or max value. The first <pctg> argument must be >=1 (greater than or equal to one) and <=500 (less than or equal to 500). If the computed percentage exceeds the range of the parameter's data type, for example, the range of a short, the range of an unsigned long, the result is silently coerced to the upper or lower bound of that type, as appropriate.

Additional syntax features and restrictions:

- Keywords (e.g., "group", "true") are case insensitive.

- In all but two cases, a <sentence> must appear on one logical input line (delimited by eol). The exceptions involve sentences terminated with open C comments or continued C strings.

- Logical input lines may be continued with '\'

- Two comment styles are supported:

  '#' delimits a comment that terminates at eol (Bourne Shell comment format)

A standard C comment, which does not nest, may be opened across more than one logical line.

**Examples**

**Selecting and Unselecting**

| Action: | <select> / <unselect> | <grpid> | <lun> | <test#> |
|---|---|---|---|---|
| Unselect all tests for COM1: | unselect | 300 | 1 | 0 |
| Unselect the MEM group's Pseudo Random Address test: | unselect | 1000 | 1 | 211 |
| Select all hard disk tests: | select | 400 | 0 | 0 |

**Setting Parameters**

| Action | <grp_param> | <grpid> | <lun> | <title> / <tag> | "=" | <data> |
|---|---|---|---|---|---|---|
| Set MEM scatter iterations at 10: | group param | 1000 | 1 | "Scatter Iterations" | = | 10 |
| Set MUL MPEG test file to spotlight.mpeg: | group param | 2500 | 1 | "MPEG Test File" | = | spotlight.mpeg" |

**Setting Metrics and Percentages**

| Action | <grp_param> | <grpid> | <lun> | <test#> | <title> / <tag> | "=" | <metric> / <pctg> |
|---|---|---|---|---|---|---|---|
| Set IDE parameter, End Sector, to percent 8, max: | test param | 2800 | 2 | 204 | "End Sector″ | = | percent (8, max) |
| Set IDE parameter, End Sector, to mid: | test param | 2800 | 2 | 204 | "End Sector" | = | mid |

## Changing the order of tests in a test script

One of the advantages to writing your own test scripts is the ability to change the order in which QA+WIN32 executes tests.  QA+WIN32 reads each script in the order that it was written hence, if you change the order of the tests in the script, you will change the order in which QA+WIN32 executes them. You can even have QA+WIN32 repeat tests later in the script. Note that script entries must follow certain syntax rules.

Note that you can **only** change the order of tests by using a text editor to manually create or edit an existing script. An easy way to do this is to use QA+WIN32 to generate a custom test script and then use a text editor such as Notepad to do the editing.

Finally, in order to run tests in the order they are written in a script, you **must** have entered the keyword "runtestsinorder = 1" at the top of the script file ('0' disables the feature) as follows:

**set runtestsinorder = 1**

Below are some examples of how to change the order of tests in a script file. The first example demonstrates how to set the Basic Operation Test of the Audio Group to run after Quick Test of the Memory Group:

```
select 1000 1 212
group param 1000 1 "Test Block Size (KB)" = 14336 /*min: 4, max: 65536  */
group param 1000 1 "Memory Test Level" = 3 /* min: 1, max: 3 */
group param 1000 1 "Test with caching enabled" = True
group param 1000 1 "Memory Allocation Method" = 2 /* min: 1, max: 3 */
select 3600 1 201
test param 3600 1 201 "Input Device" = 0 /* min: 0, max: 1 */
group param 3600 1 "Test Frequency (Hz)" = 1000 /* min: 50, max: 5000 */

group param 3600 1 "Test Level" = 16383 /* min: 3276, max: 65535 */
group param 3600 1 "Master Volume" = 32767 /* min: 0, max: 65535 */
```

The following example demonstrates how to set the Basic Operation Test of the Audio Group to run both before and after the Quick Test of the Memory Group:

```
select 3600 1 201
test param 3600 1 201 "Input Device" = 0 /* min: 0, max: 1 */
group param 3600 1 "Test Frequency (Hz)" = 1000 /* min: 50, max: 5000 */
group param 3600 1 "Test Level" = 16383 /* min: 3276, max: 65535 */
group param 3600 1 "Master Volume" = 32767 /* min: 0, max: 65535 */
select 1000 1 212

group param 1000 1 "Test Block Size (KB)" = 14336 /*min: 4, max: 65536  */
group param 1000 1 "Memory Test Level" = 3 /* min: 1, max: 3 */
group param 1000 1 "Test with caching enabled" = True
group param 1000 1 "Memory Allocation Method" = 2 /* min: 1, max: 3 */
select 3600 1 201
test param 3600 1 201 "Input Device" = 0 /* min: 0, max: 1 */
group param 3600 1 "Test Frequency (Hz)" = 1000 /* min: 50, max: 5000 */
group param 3600 1 "Test Level" = 16383 /* min: 3276, max: 65535 */
group param 3600 1 "Master Volume" = 32767 /* min: 0, max: 65535 */
```

**Note:** If you do not specify group parameters for a test and set the same test to run more than once in the script, the test parameters set for the first instance of the test will be used for all instances of that test.  Otherwise, the group parameters will override test parameters.  However, if you include different group parameters for  each instance of the test that is run more than once, the group parameters set for the last instance of that test will be used for all instances of the test.

## Running a custom test script

**To run a custom test**, do the following:

1. From the main menu, select **File/Open Custom Test Script**. The **Open** dialog box opens.

2. Navigate to the directory where the script is located. In **File name**, enter the name of the script.

3. Click **OK** to open the script. Click **Cancel** to close the dialog box without opening a script.

4. Press the  Custom Test button.

**Note:** If you have the **Stop on Fail** option selected, the custom test will stop running if it encounters a failure. If you want to run all tests, regardless of a failure, turn this selection off.  See Changing Test Options for more information.

**Note:** If your system lacks support for any tests in the custom test script, for instance if there is no audio card, then the line 'Group not available' is written to the file CstScrEr.Txt for each such test.  This can happen if you created the script on a different machine to the one you're running it on, or if you used a text editor or VQAScript.

## Using command lines

In addition to double-clicking the application icon, QA+WIN32 can be run from the command line with a number of command line options.  Open a command prompt (Select **Run** from **Start Button** on the Windows Taskbar, type 'cmd' (or 'command' on Windows 98, ME and NT), and navigate to the QA+Win32 folder) and launch the program as follows:

**QAWIN32 [options]**

where [options] are one or more of the following (Note: the program name and all command line options are case-insensitive):

| | |
|---|---|
| **/C, -C <filename>** | Sets the Custom Test script to the specified filename. This script will be run when the "Custom Test" option is selected. (Note: by default the Custom Test script is set to QA+Win32C.dat.) |
| **/Q, -Q <filename>** | Sets the Quick Test script to the specified filename. This script will be run when the "Quick Test" option is selected. (Note: by default the Quick Test script is set to QA+Win32Q.dat.) |
| **/F, -F <filename>** | Sets the Full Test script to the specified filename. This script will be run when the "Full Test" option is selected. (Note: by default the Full Test script is set to QA+Win32F.dat.) |
| **/R, -R** | Opens QA+Win32, runs the Custom Test script and then closes the application.  By default, the Custom Test script is QA+Win32C.dat; however, a different script can be run if this option is used in conjunction with /C <filename>.  For instance, |

/C myScript.dat /R

will open QA+Win32, run the script "myScript.dat" and then exit.

| | |
|---|---|
| **/DIR <directory>** | Sets the working directory to a different directory. Useful in re-routing test logs. Note: the specified directory must already exist, otherwise an 'Invalid Directory' dialog box will be displayed at start-up.  For instance, |

/DIR c:\testdir

| | |
|---|---|
| **/NoSplash** | Does not display the QA+WIN32 start up banner. |
| **/FACTORY** | Runs QA+Win32 in non-interactive mode.  The user will not be prompted for test items such as loopback plugs, and interactive tests (for instance, the mouse tests) will skip.  Can be usefully combined with /R, to run the custom script automatically in non-interactive mode. |
| **/I <grp>** | Starts QA+WIN32, only loading the group specified (where <grp> is the group name without the .dll extension). Note: to load more than one group, repeat the /I <grp> parameter, for instance: |

/I AUD /I CDN /I LPT - loads groups AUD, CDN and LPT only.

| **/O <grp>** | Starts QA+WIN32, omitting the group specified (where <grp> is the group name without the .dll extension). Note: to omit more than one group, repeat the /O <grp> parameter, for instance: |
| | /O AUD /O CDN /O LPT - loads all supported groups except for AUD, CDN and LPT |
| **/IALL** | Starts QA+WIN32, loading all groups supported by the system's hardware and software (this is the default) |
| **/OALL** | Starts QA+WIN32 without any groups. |

# Glossary

### ASCII

(**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange)

Pronounced "ask-ee."

An ASCII file is a data or text file that contains characters in ASCII format. An ASCII file can be used as a "common" format for data exchange, for example between two software applications that use their own particular data format, but are also able to convert data to and from the ASCII file format.

### AUTOEXEC.BAT file

A batch program that is run on your PC every time it is turned on. This file can contain any commands that you want to be carried out when you start your PC. Each line in this file represents a command string that you can also type at the DOS prompt.

The AUTOEXEC.BAT file is run immediately after the commands in the CONFIG.SYS file are carried out at system startup. The AUTOEXEC.BAT file is normally located in the root directory of your startup disk (usually drive C).

### BIOS

(**B**asic **I**nput **O**utput **S**ystem)

The software that controls the functioning of the input and output devices connected to the computer, such as a monitor, keyboard, mouse, or printer.

### Bus

A common channel or path for transferring data and electrical signals either internally between components in a computer or externally between devices connected in a network.

### CMOS

(**C**omplementary **M**etal **O**xide **S**emiconductor)

A type of MOS chip that provides less power consumption than standard MOS chips. It contains the "official" configuration settings for the system.

### Comma-Separated Value (CSV) file

A record layout that separates data fields with a comma and usually surrounds character data with quotes. This type of file can be imported into many different types of databases and spreadsheets.

### COM port

The **com**munication ports of the computer that allow you to communicate via modem, or to control a mouse or other serial device.  All data moves sequentially over the same wire, one bit after another. Generally slower than a parallel device, but information can be sent further over this type of connector before the onset of signal degradation. Addressed by the computer as COM1, COM2, etc.

### CONFIG.SYS file

A text file that contains commands to configure your PC's hardware components (such as the keyboard, mouse, and printer) so that the operating system and other applications can utilize them. A primary purpose of this file is to initiate the drivers for peripheral devices that have been added to the system.

This file is normally located in the root directory of your startup disk (usually drive C). When the operating system starts, it carries out the commands in the CONFIG.SYS file before the AUTOEXEC.BAT file is run.

### Configuration file

A file that contains information about a specific computer, program, user, or file.

### CSV

(See Comma-Separated Value)

### dBASE file

One of the principal database standards. Created by the Ashton-Tate Corporation, dBASE was the first comprehensive database system for personal computers.

### Device

Each of your computer's hardware components, such as the keyboard, monitor, printer, and disk drive. Each device has characteristics that can be customized.

### Device driver

A program that controls the operation of a device, often a peripheral device attached to a computer, such as a printer. The operating system includes basic drivers; other drivers may need to be added when additional devices are installed.

### Disk partition

A logical portion of a disk that organizes smaller blocks of data to be handled more efficiently and conveniently.

### DOS®

(**D**isk **O**perating **S**ystem)

A generic term for operating system. The operating system and its associated utilities provide all of the basic functions the computer and most software need to operate properly. All personal computers require some type of DOS®. The PC uses MS-DOS® or PC-DOS® or other types of DOS®.

### Interrupts

See IRQ.

**IRQ**

(**I**nterrupt **Re**Quest)

A signal that a device issues to the Interrupt controller in the PC when it wants the processor's attention, at which time the CPU temporarily suspends program execution and services the interrupt. After the interrupt has been processed, program control is again returned to the interrupted program.

**Log**

(See test log)

**NetBIOS**

A transport protocol commonly used for PC local area networks. Application programs use NetBIOS for peer-to-peer or client/server communications.

**Pass**

One complete input, processing, or testing cycle in the execution of a computer program or testing procedure.

**Path**

The location of, or route to, data on some type of storage medium. For example: the location of a file on a computer's hard disk.

In a more specific sense, path can refer to the directories and sub-directories where programs should look for particular executable files, if they are not found in the current directory. This is generally set up in an AUTOEXEC.BAT file in the root directory of the hard disk.

**RAM**

(**R**andom **A**ccess **M**emory)

The memory a computer uses to run programs.

**Serial ports**

The communication ports of the computer that allow you to hook up to a modem, mouse, printer, or other serial device(s). All data moves sequentially over the same wire, one bit after another. Generally slower than a parallel device, but information can be sent further over this type of connector before the onset of signal degradation. Addressed by the computer as COM1, COM2, etc.

**TCP/IP**

(**T**ransmission **C**ontrol **P**rotocol/**I**nternet **P**rotocol)

A set of protocols developed to allow for communication between different types of computer systems. For example, the File Transfer Protocol (FTP) provides file transfer capability, and the Simple Mail Transfer Protocol (SMTP) provides email capability.

### Test cycle

A single pass of a completed set of tests either on a single test, a group of tests under a test category, or selected tests under multiple categories. One pass is equal to one test cycle, two passes are equal to two cycles, etc.

### Test log

A file or printout, which records (or logs) the results encountered during testing.

### Test pass

(*See* Test cycle)

### Throughput

The speed with which a computer can process data. The throughput is dependent upon the configuration of the computer, which includes its internal processing capability, the efficiency of its operating system and other software, and its input/output capability.

# Eurosoft
pc reliability solutions

## Eurosoft (UK) Ltd.

Head Office
3 St. Stephen's Road
Bournemouth
Dorset BH2 6JL
United Kingdom

Tel +44 (0)1202 297315
Fax +44 (0)1202 558280
Email: info@eurosoft-uk.com
Website: www.eurosoft-uk.com

## Eurosoft (US) Inc.

North American Sales Office
Suite 101, One Davol Square
Providence, RI 02903
USA

US Toll Free Tel +1 (888) 980-9595
International Tel +1 (401) 751-9595
Fax +1 (401) 751-0088
Email: info@eurosoft-us.com
Website: www.eurosoft-us.com

### Information & Sales

| | |
|---|---|
| Europe | info@eurosoft-uk.com |
| United States | info@eurosoft-us.com |
| Pacific Rim | info@eurosoft-au.com |

### Support

| | |
|---|---|
| Europe | support@eurosoft-uk.com |
| United States | support@eurosoft-us.com |
| Pacific Rim | support@eurosoft-au.com |