Test Environment Toolkit

Release Notes for TETware Release 3.4
TET3-RN-3.4.1

Released: 19th November 1999

The Open Group

The information contained within this document is subject to change without notice.

Copyright © 1999 The Open Group
Copyright © 1996,1997 X/Open Company Limited

All rights reserved. No part of this source code or documentation may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as stated in the end-user licence agreement, without the prior permission of the copyright owners. A copy of the end-user licence agreement is contained in the file `Licence` which accompanies the TETware distribution.

Motif, OSF/1, UNIX® and the 'X' device are registered trademarks and IT DialTone and The Open Group are trademarks of The Open Group in the US and other countries.

X/Open is a trademark of X/Open Company Limited in the UK and other countries.

Win32™, Windows NT™ and Windows 95™ are registered trademarks of Microsoft Corporation.

This document is produced by UniSoft Ltd. at:

150 Minories
LONDON
EC3N 1LS
United Kingdom

Tel: +44 20 7264 2120
Email: `tet@root.co.uk`

++++++++++++++++++++++++++ TET END USER LICENCE ++++++++++++++++++++++++++

BY OPENING THE PACKAGE, YOU ARE CONSENTING TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, DO NOT INSTALL THE PRODUCT AND RETURN IT TO THE PLACE OF PURCHASE FOR A FULL REFUND.

TETWARE RELEASE 3.4 END USER LICENCE
REDISTRIBUTION NOT PERMITTED

This Agreement has two parts, applicable to the distributions as follows:

A. Free binary evaluation copies – valid for 90 days, full functionality – no warranty.

B. Free binary restricted versions – no warranty, limited functionality.

C. Licenced versions – full functionality, warranty fitness as described in documentation, includes source, binary and annual support.

PART I (A & B above) – TERMS APPLICABLE WHEN LICENCE FEES NOT (YET) PAID (LIMITED TO EVALUATION, EDUCATIONAL AND NON-PROFIT USE).

GRANT.

X/Open grants you a non-exclusive licence to use the Software free of charge if

a. you are a student, faculty member or staff member of an educational institution (K-12, junior college, college or library) or an employee of an organisation which meets X/Open's criteria for a charitable non-profit organisation; or

b. your use of the Software is for the purpose of evaluating whether to purchase an ongoing licence to the Software.

The evaluation period for use by or on behalf of a commercial entity is limited to 90 days; evaluation use by others is not subject to this 90 day limit. Government agencies (other than public libraries) are not considered educational or charitable non-profit organisations for purposes of this Agreement. If you are using the Software free of charge, you are not entitled to hard-copy documentation, support or telephone assistance. If you fit within the description above, you may use the Software for any purpose and without fee.

DISCLAIMER OF WARRANTY.

Free of charge Software is provided on an ''AS IS'' basis, without warranty of any kind.

X/OPEN DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL X/OPEN BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

PART II (C above) – TERMS APPLICABLE WHEN LICENCE FEES PAID.

GRANT.

Subject to payment of applicable licence fees, X/Open grants to you a non-exclusive licence to use the Software and accompanying documentation (''Documentation'') as described below.

Copyright © 1996,1997 X/Open Company Ltd.
Copyright © 1998,1999 The Open Group

LIMITED WARRANTY.

X/Open warrants that for a period of ninety (90) days from the date of acquisition, the Software, if operated as directed, will substantially achieve the functionality described in the Documentation. X/Open does not warrant, however, that your use of the Software will be uninterrupted or that the operation of the Software will be error-free or secure.

SCOPE OF GRANT.

Permission to use for any purpose is hereby granted. Modification of the source is permitted. Redistribution of the source code is not permitted without express written permission of X/Open. Distribution of sources containing adaptations is expressly prohibited.

Redistribution of binaries or binary products containing TETware code is permitted subject to the following conditions:

— this copyright notice is included unchanged with any binary distribution;

— the company distributing binary versions notifies X/Open;

— the company distributing binary versions holds an annual TET support agreement in effect with X/Open for the period the product is being sold, or a one off binary distribution fee equal to four years annual support is paid.

Modifications sent to the authors are humbly accepted and it is their prerogative to make the modifications official.

Portions of this work contain code and documentation derived from other versions of the Test Environment Toolkit, which contain the following copyright notices:

Copyright © 1990,1992 Open Software Foundation
Copyright © 1990,1992 Unix International
Copyright © 1990,1992 X/Open Company Ltd.
Copyright © 1991 Hewlett-Packard Co.
Copyright © 1993 Information-Technology Promotion Agency, Japan
Copyright © 1993 SunSoft, Inc.
Copyright © 1993 UNIX System Laboratories, Inc., a subsidiary of Novell, Inc.
Copyright © 1994,1995 UniSoft Ltd.

The unmodified source code of those works is freely available from `ftp.xopen.org`. The modified code contained in TETware restricts the usage of that code as per this licence.

++

1. Introduction

1.1 Preface

These release notes accompany TETware Release 3.4.

TETware includes all of the functionality of the Test Environment Toolkit Release 1.10 (TET), the Distributed Test Environment Toolkit Version 2 Release 2.3 (dTET2) and the Extended Test Environment Toolkit Release 1.10.3 (ETET), together with a number of new features.

There are two principle versions of TETware. One version is known as Distributed TETware. This version uses a client-server architecture and provides support for processing local, remote and distributed test cases. The other version is known as TETware-Lite. This version does not use a client-server architecture or require a network transport. It is more simple to set up than is Distributed TETware but does not provide support for the processing of remote or distributed test cases.

Distributed TETware is implemented on UNIX operating systems and also on the Windows NT operating system. TETware-Lite is implemented on UNIX operating systems and also on the Windows NT and Windows 95 operating systems.

Throughout this document, the Windows NT and Windows 95 operating systems are referred to collectively as **Win32 systems**. The individual system names are only used when it is necessary to distinguish between them.

1.2 Audience

This document is intended to be read by software engineers and/or systems administrators who will install TETware on their computer systems. A knowledge of system administration is assumed when TETware installation and configuration instructions are presented. In addition, a knowledge of network administration is assumed when TETware is to be built to use network transports.

1.3 Conventions used in this document

The following typographic conventions are used throughout this document:

- `Courier font` is used for function and program names, literals and file names. Examples and computer-generated output are also presented in this font.
- The names of variables are presented in *italic font*. You should substitute the variable's value when typing a command that contains a word in this font.
- **Bold font** is used for headings and for emphasis.

1.4 Related documents

Refer to the following documents for additional information about TETware:

- *Test Environment Toolkit: TETware Installation Guide*
  There is one version of this document for each operating system family on which TETware is implemented.
- *Test Environment Toolkit: TETware User Guide*

- *Test Environment Toolkit: TETware Programmers Guide*
- *Test Environment Toolkit: TETware Knowledge Base*

Source and postscript versions of these documents are included in the TETware distribution.

1.5 Problem reporting

If you have subscribed to TETware support and you encounter problems when building or executing TETware you should take a copy of the error reporting template contained in the file *tet-root*`/doc/tet3/err.template` in the distribution, fill in the details of the problem, and send it by electronic mail to:

`tet_support@opengroup.org`

Alternately you may use a web browser to complete and submit the HTML form included in the file *tet-root*`/doc/tet3/errtemp.html` in the distribution.

You should include sufficient information in your report to enable someone who is unfamiliar with your system to be of assistance in solving the problem.

Users are reminded that the user-contributed software which accompanies the TETware distribution is not covered by TETware support services.

2. New features in this release

The following features appear for the first time in this release of TETware:

- The Java API has been integrated into the TETware Distribution. It is supported on Linux, Solaris and Win32 platforms.
- An example Java test suite has been added to the distribution.
- Chapters describing the Java API and the example test suite have been added to the Programmers Guide.
- It is now possible to instruct `tcc` to write the journal to the standard output or to a pipeline.
- A work-around for a bug in `ksh93`'s file descriptor handling in a subshell has been added to the Korn Shell API.
- The support for POSIX threads in the Thread-Safe C and C++ APIs has been enhanced. Refer to the section entitled ''Enhanced support for POSIX threads'' later in these Release Notes.
- A number of problems reported after the previous TETware release have been fixed. All the source code patches issued since the last TETware release have been incorporated into this release.
- The documentation has been revised and updated. A number of articles have been added to the TETware Knowledge Base.

3. Status of this release

This release of TETware is a general release for production use.

4. Problems fixed since the last release

The following problems have been fixed since the last TETware release:

- In the source distribution, `make clobber` did not remove the compatibility links that were created using `make compat`. A new target called `compat_clean` has been added to the top-level makefile to perform this task.
- It was possible for `tcc -i` to overwrite an existing journal file.
- Configuration variable expansion did not work correctly in TETware-Lite.
- On UNIX systems the library function that checks file access permissions did not check group permissions against the user's supplementary group list.
- During a configuration variable exchange, `tccd` would die on systems where `glibc` is implemented.
- If the system time was altered by a significant amount while `tcc` was running, this could confuse the timeout mechanism in `tcc`'s execution engine.
- The Korn Shell API did not correctly handle a user-defined result code name that contained embedded white space.
- When `tcc` was invoked with the `-v` option and the `TET_RUN` environment variable was set, `tcc` would sometimes fail with an assertion error while processing configuration variables.
- An error message was printed unnecessarily when `tcc`/`tccd` performed a recursive copy of an empty directory.
- When a test case called `tet_remexec()`, the API only looked for the `tetexec.cfg` file in the test suite root directory on the target system. Now, if an alternate execution directory is defined on the target system, the API looks for the `tetexec.cfg` file there first, and only looks in the test suite root directory if the file cannot be found in the alternate execution directory.
- The scenario parser in `tcc` used to permit a test case name to contain a `..` component. This caused problems later on when a test case was executed with `TET_EXEC_IN_PLACE` set to False, or when Saved Files processing was performed.
- When the Thread-safe C and C++ APIs were used in conjunction with a POSIX threads implementation which re-used a thread ID immediately the corresponding thread had been joined, this could result in the TCM's thread cleanup code waiting for the wrong thread.

5. Known problems in this release

There are no known problems in this release.

6. Building and installing TETware

6.1 Building and installation instructions

For information on how to build and install TETware, please following the instructions in the version of the TETware Installation Guide which is appropriate for your system.

Note: There have been some changes to the makefile scheme in order to support the Java API. All the `defines.mk` files that are supplied with the distribution have been updated to support the new scheme. However, if you use your own `defines.mk` file when building TETware, you will need to add an extra variable assignment to the file before you can use it with the new scheme.

You should add the following variable assignment in order to enable your `defines.mk` file to continue to work with the new makefile scheme:

```
JAVA_COPTS = JAVA_NOT_SUPPORTED
```

The new makefile scheme also uses a command to perform partial linking. If you use your own `defines.mk` file you will need to add a variable which defines the command to use. On many UNIX systems it would look something like this:

```
# LD_R - the program that performs partial linking
LD_R = ld -r
```

Note: In this release the Java API is supported for use with JDK Version 1.1 on Solaris, Linux and Win32 systems. In order to build the Java API on these systems a variable must be set in the `defines.mk` file which specifies where the Java Development Kit (JDK) has been installed on your machine.

On Solaris Release 2.7 the JDK is supplied with the operating system and so is installed in a standard place (`/usr/java`). This location is specified in the `defines.mk` file for Solaris 2.7 that is supplied in the distribution.

However, on other machines the JDK might be installed anywhere, so it is necessary to customise your `defines.mk` file if you want to build the Java API. Refer to the section entitled ‘‘Support for Java’’ in the TETware Installation Guide for instructions on how to do this.

6.2 Installed platforms

6.2.1 UNIX systems

Both TETware-Lite and Distributed TETware versions have been installed and tested on the following platforms:

- AIX Release 4.3.1 using `c89`
- HP-UX Release 11.00
- Linux version 2.0.35 (Slackware) using `gcc` version egcs-2.90.29 and `libc5`
- Solaris Release 2.7 using the Sun Workshop Compiler

- UNIX System V Release 4
- UnixWare Release 2.1.1

6.2.2 Win32 systems

TETware-Lite and Distributed TETware have been installed and tested on Intel PCs running Windows NT release 4.0.

TETware-Lite has been installed and tested on Intel PCs running Windows 95.

6.3 API status

6.3.1 Thread-safe APIs

The Thread-safe APIs have been exercised on the following platforms:

- Linux using POSIX threads
- Solaris using POSIX threads and UI threads
- UnixWare using UI threads
- Win32 systems

The following behaviour has been observed when using the Thread-safe API on UnixWare:

i. A call to `tet_printf()` in a non-main thread causes the process to receive a `SIGSEGV` signal.

ii. Cleanup of left-over threads sometimes causes a fatal error.

It is believed that this behaviour may be due to a problem with the threads implementation on this platform.

On Win32 systems the thread-safe APIs must be used with the multi-threaded DLL version of the C runtime support library. Use with the multi-threaded static version of the C runtime support library is not supported.

6.3.2 C++

The C++ APIs have been exercised on Linux, Solaris, UnixWare and Win32 systems.

6.3.3 Java

The Java API is only supported when used with JDK v1.1 on the following platforms:

- Linux using ‘‘green’’ threads
- Solaris using native threads
- Win32 systems

6.4 Transport-specific status

6.4.1 Socket network interface

Versions of Distributed TETware using the socket network interface have been tested on all the platforms listed previously.

6.4.2 XTI network interface

Versions of Distributed TETware using the XTI network interface have been tested on AIX 4.3, Solaris 2.7 and UnixWare 2.1.1 using TCP as the underlying transport provider.

There is a known problem with some SVR4 XTI implementations in which the `t_sync()` function does not work correctly. An XTI implementation may hold transport endpoint data either in kernel or in user address space. The `t_sync()` function is included in the XTI specification for the benefit of implementations which do not automatically detect when the transport endpoint data held in user space is lost. Examples of when such loss might occur are when the file descriptor underlying a transport endpoint is duplicated using `fcntl()` or when the process address space is overlaid by one of the `exec()` system calls.

Since the XTI version of TETware may perform both of these operations, it will not function when used with an XTI implementation which does not automatically detect the loss of data held in user address space and in which `t_sync()` does not perform the advertised function. A common symptom of this problem is when one of the servers `tetsyncd` and `tetxresd` fails with a `TBADQLEN` error associated with a `t_listen()` call soon after being started by `tcc`.

The XTI version of Distributed TETware has not been tested using an OSI connection orientated transport provider.

7. Enhanced support for POSIX threads

The behaviour of the POSIX thread-safe C API library has been changed in this release in order to conform more strictly to the requirements of the POSIX threads standard.

Details of API build options are presented in the section entitled ‘‘Support for Threads’’ in Chapter 3 of the TETware Installation Guide for UNIX Operating Systems. Details of API issues are presented in the sections entitled ‘‘Use of API functions in child processes’’ and ‘‘`tet_thr_join()` and `tet_pthread_join()`’’, both in Chapter 10 of the TETware Programmers Guide.

8. TETware for TET and ETET users

Some hints and tips for users experienced with previous TET implementations are presented in an appendix to the TETware User Guide.