The Linux Tcl and Tk HOWTO

Table of Contents

Table of Contents

The Linux Tcl and Tk HOWTO

Luca Rossetti lukaros@tin.it

v0.2, 07 November 1998

This document describes the Linux approach to Tcl, a scripting langua ge. It is an easy to learn interpreted language that uses a typeless approach to achieve a higher level of programming and a rapid application development. The Tk toolkit is a programming environment for creating graphical user interf aces (GUI) under X Window System. Their capabilities include the possibility to extend and embed in other application, rapid development and ease of use. Toge ther, Tcl and Tk provide many benefits both to application developer and user. Tk−ba sed interfaces tend to be much more customizable and dynamic than those built with one of the C or C++ based toolkits. Tk implements the Motif look and feel. A great number of interesting X applications are implemented entirely in Tk, with no new application−specific commands at all.

1.Introduction

2.Tcl/Tk History

3.What is Tcl/Tk

4.Installing and getting started with Tcl and Tk

5.Just a little bit of Programming

6.Scripting Language: pros and cons

7.Most Famous Programs using Tcl/Tk

8.References

9.Tcl/Tk License Terms

1.Introduction

This is the Linux Tcl and Tk HOWTO. It is intended as a Linux reference covering everything you should know concerning installation, configuration and an introduction to development under Tcl and Tk. History and some pros and cons about Tcl and Tk under Linux are analized, and references are given to many other sources of information on a variety of topics related to this simple but powerful scripting language.

If you ever rebuilt your Linux kernel inside X using the command

```
make xconfig
```

you surely managed to face the strenght of this simple but powerful scripting language.

After executing the first step of kernel rebuilding, a script called kconfig.tk is executed via wish (the Tcl intepreter). The Linux Kernel Configuration wind ow appears. Instead of answering a series of questions, this X−based configuratio n utility allows you to use checkboxes to select which kernel options you want to enable.

The system stores your last configuration options so that every time you run it, you need only to add or remove some checks and you don't need to reent er the whole set of option. Moreover you can fill the whole (or part of the) list of kernel option the order you want. After this simplified step you can rebuil d your kernel in the traditional way.

There's actually another famous case. Have you ever used "printtool" ? (Printer Config Tool (C) Copyright 1994 by Red Hat Software − author: Michael Callahan). If you installed a Red Hat distribution you happily managed to use it to set up printing services . Well, printtool front−end is mainly a Tcl/Tk script.

For those who don't know Red Hat let me tell you how you can easily configure your printers just filling some textboxes and filling in some checkboxes.

The program itself takes care of setting up printing services through the creation of spooling directory, writes /etc/printcap file and the printer's filter, reloads lpd and tests your filter with ascii or postscript pages. It allows you manipulate ghostscript options (i.e. choose up to 8 pages per outpu t page and setting margins), has an help−on−line and many more features.

What is the difference compared to other service−printing implementation?

Everything is achieved by using Tcl/Tk as a "glue" between consolidated application and operating with normal Linux configuration files in a visual and interactive window under X−Window. No new application−specific commands were written at all.

1.1 Purpose of this document

Currently the purpose of the document is limited to giving initial references to Linux users; in future versions I will try to incorporate a small "Programm ing Tutorial". Let me state that again: this is not (and doesn't want to substitute) an omnicomprehensive "user manual" or "reference" for Tcl and Tk development and programming − it's just a starting point for Linux users.

The author's concept of reference manual coincides with the definition of man pages and many people learn Tcl/Tk from these basic source of informati on. These files are a part of the source code distribution and are installed on your Linux box. You will be able to access the Tcl/Tk manual pages through the man command.

Many structured and complete programming tutorials have been written in order to let the new user begin hacking with Tcl/Tk; tons of other material are available in the Internet. Interpreting Tcl/Tk philosophy I won't try to reinvent the wheel, I will manage to glue the huge amount of material already available.

I suggest you to have a look at the other documents listed in the References section in order to find where to retri eve specific information about Tcl and Tk.

1.2 What you should know before reading

In order to understand the abc of Tcl/Tk you shouldn't be a programming−guru, command sintax is very simple. Basically you have to be familiar with:

- simple programming concepts;
- using very common unix commands and/or utilities;
- having access to the Internet;
- using ftp.

1.3 New versions of this document

Newer versions of this document will be uploaded to ftp site: ftp://sunsite.unc.edu/ pub/Linux/HOWTO/ and will be available on all other mirrors.

Hypertext and other versions of this and other Linux HOWTOs are available mainly at sunsite.unc.edu/LDP/lpd.ht ml and www.linux−howto.com and on many other Mirror World−Wide−Web sites.

I will try to incorporate in my Web−Page the most recent html and sgml version of the document. Most Linux CD−ROM distributions include the HOWTOs, often in a subdir of /usr/doc/ directory, and you can also buy printed copies from several vendors.

Sometimes the HOWTOs available from CD−ROM vendors, ftp sites, and in hardcopy format are out of date. If the date on this HOWTO is more than six months in the past, then a newer copy is probably available on the Internet.

If you make a translation of this document into another language, drop me a line and I'll include a reference to it here.

1.4 Revision History

- **Version 0.1**: 28 October 1998 − first version;
- **Version 0.2**: 07 November 1998 − deep restyling and lifting.

1.5 Feedback and other stuff

I rely on you, the reader, to make this HOWTO useful. If you have any suggestio ns, corrections, or comments, please send them to me, (Luca Rossetti), and I will try to put them in the next revision.

If any of the links mentioned in this document becomes unavailable or changes, please notify me immediately so that I can update and/or remove the link.

I am willing to answer general questions about Tcl/Tk and Linux as best as I can. Before doing so, please read all of the information in this HOWTO, and send me detailed information about the problem.

If you publish this document on a CD−ROM or in hardcopy form, a complimentary copy would be really appreciated. E−mail me and I will send you back my postal address.

In many section I mention publishing companies or commercial URL sites: I really don't work for these guys.

1.6 Credits

Most of the information in this HOWTO comes from Dr. Ousterhout's Scriptics and Larry W. Virden comp.lang.tcl FAQs .

I would like to thank the PLUTO Italian Linux User Group and the whole volunteers of ILDP (Italian Linux Document ation Project) especially Eugenia Franzoni and Giovanni Bortolozzo for their feedbac k.

1.7 Distribution Policy

This document is Copyright 1998 by Luca Rossetti.

This document is distributed in the hope that it will be useful to the reader: of course it is considered to be without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose. This HOWTO is free documentation; you can redistribute it and/or modify it under the terms of the LDP COPYRIGHT .

Please read the LDP Manifesto for more details.

2.Tcl/Tk History

Tcl/Tk originated with Dr. John Ousterhout (pronounced "Oh'−stir−howt") while teaching at University of California, Berkeley, Califor nia. He actually started implementing it when he got back to Berkeley in the spring of 1988; by summer of that year it was in use in some internal applications, but there was no Tk. Read about the history of Tcl/Tk directly from its author 's words at www.scripti cs.com/scripting/tclHistory.html.

2.1 Cronology

- **1989:** The first external releases of Tcl and beginning of Tk imp lemention;
- **1991**: First release of Tk;

- **1994:** Dr. Ousterhout was hired by Sun Microsystems, Inc.: he was a Distinguished Engineer and led the Tcl project.
- **April/May 1997:** the Sun research group responsible for Tcl devel opment were spun off into a Sun business group called SunScript. However, things chan ged again soon afterwards. You can read more about that evolution selecting "SunSc ript_story" at URL: Su nScript−Story .
- **August 1997:** a Tcl Consortium was formed. You can read more abou t it, including how to join, who currently serves on the board of directors, and so on by visi ting Tcl Consortium;
- **February 1998:** Dr. Ousterhout left Sun to create Scriptics, a company dedicated to scripting tools, applications, and services. According to www.scriptics.com /about/news/qa.html, core Tcl and Tk remain free, with the team at Sun continuing work right now on Tcl/Tk 8.1. After the next release, the intention is that work on the core will migrate from Sun to Scriptics, with the Sun team will focus more on Tcl extensions and applications.
- **April 23, 1998:** the Association for Computing Machinery ACM awarded the 1997 Software System Award to John Ousterhout and Scriptics (www.acm.org/awards/). This award is given to an institution or individual recognized for developing a software system that has had a lasting influence, reflected in contributions to concept s, in commercial acceptance, or both.

3.What is Tcl/Tk

3.1 What is Tcl?

Tcl is the acronym for "Tool Command Language" (it is pronounced "tickle"). Tcl is actually divided into two things: a language and a library.

Tcl is a simple textual programming language, intended for issuing commands to interactive programs such as text editors, debuggers and shells. It has a simple syntax and it is also programmable.

Tcl users can write command procedures to provide more powerful commands than those given in the built−in set.

Second, Tcl is a library package embeddable in applications. The Tcl library consists of a parser for the Tcl language, routines to implement the Tcl built −in commands, and procedures which allow each application to extend Tcl with addit ional commands specific to that application. The application program generates Tcl commands and passes them to the Tcl parser for execution.

Commands may be generated by reading characters from an input source, or by associating command strings with elements of the application's user interfa ce, such as menu entries, buttons, and other widgets. When the Tcl library receive s commands it parses them into component fields and executes built−in commands directly.

For commands implemented by the application, Tcl calls back to the application to execute the commands. In many cases commands will make recursive invocation s of the Tcl interpreter by passing in additional strings to execute (in fact procedures and conditional−looping commands all work in this way). An applicat ion program can obtain many advantages by using Tcl for its command language:

- Tcl provides a standard syntax: once users know Tcl, they will be able to issue commands easily to any Tcl−based application.
- Tcl succeeds to provides programmability. All a Tcl application needs to do is to implement a few application−specific low−level commands. Tcl provides many utility commands and a general programming interface for building up comp lex command procedures. By using Tcl, applications need not reimplement these feat ures.
- Extensions to Tcl, such as the Tk toolkit, provide mechanisms for communi cating between applications by sending Tcl commands back and forth. The common Tcl language framework makes it easier for applications to communicate with one another.

It is important to note that Tcl was designed thinking that the programmer should actually use two or more languages when designing large software system s. One for manipulating complex internal data structures, or where performance is important, and another, such as Tcl, for writing very small scripts that glue together the other pieces, providing hooks for the user to extend.

For the Tcl script writer, ease of learning, ease of programming and ease of gluing are more important than performance or facilities for complex data structures and algorithms.

Tcl was designed to make it easy to drop into a lower language when you come across tasks that make more sense at a lower level. In this way, the basi c core functionality can remain small and one need only bring along pieces that one particular wants or needs.

One answer to "What is Tcl?" can be found at www.NeoSoft.com/tcl /whatistcl.html .

3.2 What is Tk?

Tk (pronounced "tee−kay") is an extension to Tcl which provides the programmer with an interface to the X11 windowing system . Note that Tk has been successf ully compiled under X11 R4, X11 R5, X11 R6, as well as Sun's NeWS/X11 environments.

Many users will encounter Tcl/Tk via the "wish" command. Wish is a simple windowing shell which permits the user to write Tcl/Tk applications in a proto typing environment.

At present Tcl/Tk cannot handle Japanese, Chinese, Korean, ... language fonts.

3.3 Extensions

Since Tcl is so easy to extend, many try to share extensions, including the popular itcl, [incr Tcl], ObjectTcl, TclX, Tix , and BLT.

These extensions, of course, require an extended Tcl interpreter. Moreover, many Tcl free applications require a particular Tcl extension to run.

One of the most popular extension is called Expect. It allows you to place a friendly front−end inside most command−line based UNIX applications, such as ftp, telnet, rlogin, passwd, fsck, and so on.

A complete list of Tcl/Tk extensions can be found at URL www.scr iptics.com/resource/software/extensions/.

3.4 Supported Platforms

This section contains information about Tcl 8.0 and Tk 8.0, the most recent version of Tcl/Tk. They were originally released on August 18, 1997 and the most recent patch releases (8.0.3) were made on September 3, 1998.

When you download Tcl and Tk you get two programs, wish and tclsh, supporting script libraries, and on−line reference documentation. These programs are gene ral purpose platforms for writing applications with Tcl. Wish includes the graphic al user interface toolkit Tk. The packages are ready to use after installation.

Tcl 8.0 and Tk 8.0 run on most releases of the following operating systems:

- Windows 95
- Windows NT
- Solaris and SunOS
- Linux
- HP−UX
- SGI
- IRIX
- Digital Unix
- AIX
- SCO Unix
- Most other Unix−like operating systems Macintosh (68K and Power Mac)
- Pre−compiled releases are available for different Linux distribution.

4.Installing and getting started with Tcl and Tk

Most modern distribution include Tcl and Tk. Rpm and deb packages with precompiled binaries are avalaible for Red Hat, SuSE and Debian distributions (that'll make installation easier).

A modern distribution standard installation will include Tcl/Tk as it is needed by many configuration tools running mainly under X.

Tcl and Tk are distributed freely in source form via the Internet. There are no restrictions on their use and no licenses or royalties are needed (see the license.terms section for complete information).

Many more Tcl/Tk scripts and extensions are also available freely.

4.1 Downloading the Core Distributions

The Tcl/Tk core consists of the Tcl and Tk libraries, plus the wish and tclsh applications, associated documentation, script libraries, and demonstrat ive applications. The primary FTP site for this information is ftp.scriptics.com/pub/tcl/.

The primary HTTP site is www.scriptics. com/software/download.html .

4.2 Installation

Unless already available for your Distribution in proprietary packages you'll want to download the source release. You'll need both Tcl and Tk source s. This procedure refers to the second case.

Choose between compressed tar and gzipped tar format.

Compressed Tar Files

Tcl sources (tcl8.0.3.tar.Z): compressed tar file (about 2.4 Mbytes). Tk sources (tk8.0.3.tar.Z): compressed tar file (about 3.3 Mbytes).

Gzipped Tar Files

Tcl sources (tcl8.0.3.tar.gz): gzipped tar file (about 1.5 Mbytes). Tk sources (tk8.0.3.tar.gz): gzip'ed tar file (about 2.1 Mbytes).

When you retrieve one of these files, you will get a compressed tar file with a name like tcl8.0.3.tar.gz or tcl8.0.3.tar.Z. The files are identical except for the technique used to compress them (.gz files are generally smalle r than .Z files).

To unpack the distribution, invoke shell commands like the following, depending on which version of the release you retrieved:

```
gunzip -c tcl8.0.3.tar.gz
```

or

```
tar xf - zcat tcl8.0.3.tar.Z
```

or

```
tar xf - unzip tcl80.3.zip
```

Each of these commands will create a directory named tcl8.0.3, which includes the sources for all platforms, documentation, and the script library for Tcl 8.0. To compile and install the distribution, follow the instructions in the README file in the distribution directory. Be sure to compile Tcl before Tk, since Tk depends on information in Tcl.

4.3 The Contributed Archive

There are many other freely available packages for Tcl and Tk, including both scripts written in Tcl and extensions written in C or C++. These packages include database applications and network access, a graphical user interface builder, the expect program, additional Tk widgets, and dozens of other things . The primary site for the Tcl/Tk archive is ftp.neosoft.com/pub/tcl.

4.4 Mirror Sites

Several other sites around the world mirror the whole or part of the material from the core site and the contributed archive; you may find more useful to retrieve information from a mirror site that is close to you.

Ftp file "0_mirror" at: ftp.scriptics.com/pub/tcl/ for a list of the mirror sites in your country.

4.5 Which Releases to Use

Always refer to newer recommended version in section "Tcl/Tk Core" of the

Scriptics Soft ware Central page.

At the time of this writing, recommended releases are the latest (Tcl 8.0.3 and Tk 8.0.3), which were released in September, 1998. Tcl 8.0 contains a new bytecode compiler that can speed up execution by a factor of 2−10x. It also provides namespaces, binary I/O, and several other new features.

Tk 8.0 is the first release to provide native look and feel on Macintoshes and PCs. Tk 8.0 also supports application embedding and has a new portable font mechanism. Both Tcl 8.0 and Tk 8.0 provide additional features in the Safe−Tcl security model.

4.6 Where to report problems, bugs, or enhancements

Use comp.lang.tcl for public communicati ons.

The alternative would be to send problems, suggestions, new ideas, etc. directly to the author. Email to John Ousterhout will reach the author of Tcl and Tk.

When reporting problems or bugs, be sure to mention all the details needed for a correct diagnosis. Basically you will have to describe what hardware, operating system and version of Tcl/Tk you are using, if you have made any modification or add on and provide, if possible, either a small piece of code, or a URL to some code which demonstrates the problem.

If you have software from which you think the community might benefit (either a program, function, extension, or simple example), or you have a document, magazine or journal article, thesis, project, or even commercial advertisement , be sure to let the appropriate guys know.

There are FAQ maintainers for each of these areas as well as a comp.lang.tcl.announce newsgroup you can use.

It is always worthwhile to submit your contributions directly to the ftp site so more folk in the future can benefitthanks to your experience.

To make announcements to the comp.lang.tcl.announce newsgroup, send email with the details to tcl−announce. Also, feel free to just point folk at your own ftp site or WWW site if you have one which can be used .

5.Just a little bit of Programming

Since Tcl is an interpreted language, to run a Tcl program (also called a script), you normally pass the script file to the Tcl interpreter, wish, for example:

```
wish hello.tcl
```

You can also use wish in interactive mode and type in commands at the command line.

There's another standard Tcl interpreter, tclsh, which only understands the Tcl language. Tclsh does not have any of the Tk user interface commands, so you cannot create graphical programs in tclsh.

Some Tcl freeware applications extend the Tcl language by adding new commands written as C functions. If such is the case, you need to compile the applicati on instead of just passing its Tcl code to the wish interpreter. This application program, from a Tcl perspective, is really a new version of the wish interpret er, which the new C commands linked in. Of course, the application program may be a lot more than merely a Tcl interpreter. (Note: you can also use Tcl's auto−loading capability on systems that support it.)

5.1 The one−minute program in Tcl

Tcl has a simple structure. Each line starts out with a command, such as button and a number of arguments. Each command is implemented as if it was a C function. This function is responsible for handling all the arguments.

As a very standard example, the following is the Hello World program in Tcl/Tk:

```
# This is a comment
button .b −text "Hello World" −command exit
pack .b
```

In this case you have to type the commands interactively to tclsh or wish.

You can also place command into script files and invoke these just like shell scripts. To do this for the previous example, rewrite the Hello World program as follows:

```
#! /usr/local/bin wish -f
button .b -text "Hello World" -command exit
pack .b
```

Put the text inside a file called Hello and make sure that wish is installed in /usr/local/bin (otherwise you will have to change opportunely the path).

Make the file Hello executable issuing the command

```
chmod 775 Hello
```

and run it inside X.

You will see a button labelled Hello World inside a window: clicking it will close (exit) the window.

6.Scripting Language: pros and cons

To understand the importance and the future of Tcl/Tk I strongly suggest to point your web browser at the URL www.scriptics .com/people/john.ousterhout/ by John K. Ousterhout . You will read about the importance and the comparison between scripting (in langua ges such as Tcl) and system programming (in languages such as C and Java).

To read a document about comparisons see the comparison discussion.

Here's a summary of the most important pros and cons about Tcl/Tk.

6.1 Some of the most common complaints about Tcl

Tcl is interpreted

The data is primarily treated as strings, programs written in Tcl are slow. Tcl 8.x attempts to address this by doing some degree of compilation as well as permitting additional variable types.

Several characteristics are not intuitive

Comments are commands rather than traditional comments, numbers beginning with 0 are octal, proper use of quoting mechanisms, etc. These aspects are covered in the various FAQs.

6.2 Some of the most pros about Tcl

It is a high−level scripting language

You need to write a lot less code to get your job done, especially when compared to Motif or Win32 applications. In general, the number of Line Of Code (LOC) of a software project is one of the most important complexity index es.

Tcl is free

You can get the sources for free over the Internet from Scriptics Download Page or from the FTP site for Tcl. The software c ore site includes the source code version, as well as binary versions for Windows and Macintosh platforms; or, you can get Tcl on a number of CD−ROMs for a nominal cost.

Read about Tcl and Tk core free at www.scriptics.com/about/n ews/qa.html .

It runs on many platforms

Versions exist for UNIX (Linux... of course), Windows and Macintosh. Except for a few platform differences, your Tcl scripts will run the same way on all systems.

It is interpreted

You can execute your code directly, without compiling and linking (though Tcl compilers are available).

It is extensible

It's easy to add your own commands to extend the Tcl language. You can write your commands in C or Tcl.

It is embeddable in your applications

The Tcl interpreter is merely a set of C functions that you can call from your code. This means you can use Tcl as an application language, much like a macro language for a spreadsheet application.

Tcl/Tk is Year 2000 (Y2K) compliant

Read what the creator of the Tcl and Tk core tells about this topic www.scriptics.com/Y2K.html .

7.Most Famous Programs using Tcl/Tk

Apart from the two implementation described in the Introduction, there are many applications writte n in Tcl/Tk or a combination of Tcl and C. A complete list where to look for these implementation is the part 4 of the frequently−asked questions on Tcl/Tk (FAQs). I suggest you to visit

Scriptics' Softwar e Central.

Another good starting point is www.NeoSoft.com/tcl/.

One of the best Tcl applications running under Linux is called TkDesk a window manager and application launcher that works very well.

If you're a Tcl/Tk Linux developer, feel free to send me a URL (and a description of the work) that I can link to in here.

8.References

For many reasons people often like having a hard−copy manual as a reference or like to be helped by other folks online.

Here you can find a selection of reference books, tutorials, www−sites and newsgroup.

8.1 Books

Many books concerning Tcl/Tk were written and are to be published. I won't even try to list them all (**another howto woudn't be enough for that** :)). You can find much more information plus additional notes at:

- www.tcltk.com/consortium/resources/books.html
- www.amazon.com/exec/obidos/subst/categories/computer−programming/tcl−tk−article/002−8989352−4516417
- Page: tcl_books.html at URL starbase−neosoft−tcl−books

Here I will try to summarize with some notes the book I know concerning the subject, they're all at a basic−medium level. Again, people who know the subject, have enough information about where to find advanced level books.

Tcl and the Tk Toolkit

Author: John K. Ousterhout

WWW book information: cseng.aw.com/bookdetail.qry?ISBN=0%2D201%2D63337%2DX&ptype=0

Book's examples: ftp.script ics.com/pub/tcl/doc/book.examples.Z

Book suppliment: www.scriptics.com/doc/tk 4.0.ps

The book primarily covers Tcl 7.3 and Tk 3.6.

Practical Programming in Tcl and Tk, II ed.

Author: Brent Welch

WWW book information: www.beedub.com/book/

Book's table of contents: www.beedub.com/book/

Book promotion info at section_50000.html of the URL www.borders.com/sections/

This updated edition describes Tcl / Tk 8.0 as it was during the beta period. Along with the material from the first edition, it also covers sockets, packag es, namespaces, a great section describing the changes in Tcl 7.4, 7.5, 7.6, and 8.0 (and Tk as well), Safe Tk and the Plugin!

Tcl and Tk Reference Manual

Editors: Donald Barnes, Marc Ewing, Erik Troan

WWW book information: www.lsl.com/catalog/bo oks/tcltk/

The Visual TCL Handbook, 1/e

Author: David Young

WWW book information: www.amazon.com/exec/obidos/ISBN=013461674X/u/7141−5908756−107481

A comprehensive guide to Visual TCL. This book leads reader from basic graphical user interface development concepts to meaningful application develo pment. The book focuses on the TCLX and VT extensions, addressing many fundamental TCL topics. VT is a Motif based graphical interface, incompatible with Tk. The entire TCL language is documented in a separate Commands section. Comes with a CD−ROM that includes SGI, Solaris, HP−UX, AIX and Unixware versions of Visual Tcl.

Running LINUX

Author: Matt Welsh and Lar Kaufman

WWW book information: www.ora.com/ca talog/runux2/noframes.html

Running Linux is a really well written basic book. It has a chapter on programming using Tcl/Tk. (and Perl, C, C++).

Tcl/Tk for Dummies (For Dummies)

Author: Timothy Webster, with Alex Francis

WWW book information: www.dummies.com/

Another one of the series of the paperback programming books. This one focuses on the Tcl plugin as a programming environment.

Interactive Web Applications With Tcl/Tk

Authors: Michael Doyle Hattie Schroeder

WWW book information: www.eolas.com/tcl/

This is a learning by example book, for people who know a bit of programming, but are not experts. It covers developing applets as well as stand−alone appli cations and simple server applications. The book comes with the Spynergy toolkit, whic h adds a variety of pure Tcl/Tk procedures for distributed processing, URL retri eval, HTML rendering, database management and platform independent file managment, Ed, a Tcl editor and testing environment, an image conversion tool, a demo of Tk features, a client/server version of a rolodex application, a pure Tcl web server, a client/server push application, a tcl web browser,

8.2 Manual and On−line Tutorials

- John Ousterhout has written an engineering style guide that describes the coding, documentation, and testing conventions that will be used at Sun in the coding of the C code in the Tcl core and has made it available to other Tcl and Tk developers. It is located at ftp.script ics.com/pub/tcl/doc/engManual.tar.Z
- A second style guide, covering the writing of Tcl scripts, can be found at ftp.scr iptics.com/pub/tcl/docs/styleGuide.tar.gz. Other versions of it can be found at sunscript.su n.com/techcorner/styleGuide.ps.
- A brief introduction to TCL/TK by David Martland
- Another tutorial untitled User interfaces with Tcl/T kwas written by Fintan Culwin .
- Although you should have your Tcl/Tk manual pages on your system, here's another place where to look for TCL Manual Pa ges (from TCL7.4)−Tk Manual Pages (from TK4.0).

- The Tcl/Tk Cookbook offe rs a lot of getting−started information.

8.3 World Wide Web sites

There are a great number of WWW resources which provide additional information about many aspects of Tcl and its extensions.

- Refer to Tcl−FAQs (pa rt2) for a great number of pointers to Tcl/Tk documentation and web sites.
- Point your web browser at Tcl/Tk Information : a site with many links to TclTk resources on the web (Information, Extensions, Tools, Training and Events).
- Have a look at Tcl−Consortium : a non−profit organization which promotes Tcl/Tk and supports the development community.
- Point your Web browser at The Official Contributed Sources A rchive for the Tool Command Language (Tcl) and the Tk Toolkit, hosted by NeoSoft, Inc.
- For a discussion dealing with the pros and cons of the major scripting languages : article in SunWorld by Cameron Laird
- A web page that contains a variety of comparisons between Tcl/Tk and othe r similar systems. Most of them are taken from "comp.lang.tcl", the author would be happy to add any other important article that you folks want to send to him.: Comparison by Wayne Christopher.
- E.J. Friedman−Hill's Tcl/Tk Course: this document is available only in PowerPoint source form and in low−quality HTML form (in HTML format all the figures and some of the text is missing).

8.4 Other documents & Frequently Asked Questions

A lot of material is available on the Internet: introductory papers, white papers, tutorials, slides, postscript versions of published books in draft and many more.

For a complete reference please give a look at the excellent Tcl−FAQs .

8.5 Newsgroup

comp.lang.tcl is an unmoderated Usenet newsgroup, created for the discussion of the Tcl programming language and tool s that use some form of Tcl, such as the Tk toolkit for the X window system, Extended Tcl, and expect.

For Tcl/Tk related announcements always refer to comp.lang.tcl.announce : you will find release announcement, patches, new application and so on.

Again, faq could be retrieved at Tcl−FAQs.

9.Tcl/Tk License Terms

The following terms apply to the all versions of the core Tcl/Tk releases, the Tcl/Tk browser plug−in version 2.0, and TclBlend and Jacl version 1.0. Please note that the TclPro tools are under a different license agreement. This agreement is part of the standard Tcl/Tk distribution as the file named "license.terms".

TCL/TK LICENSE TERMS

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and other parties. The followin g terms apply to all files associated with the software unless explicitly discla imed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON−INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behal f of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227−7013 (c) (1) of DFARs. Notwithstanding the foregoing , the authors grant the U.S. Government and others acting in its behalf permissi on to use and distribute the software in accordance with the terms specified in this license.
