The Tcl Programming Language

A Comprehensive Guide

Ashok P. Nadkarni

The Tcl Programming Language: A Comprehensive Guide

Ashok P. Nadkarni

The `asciidoctor` and `asciidoctor-fopub` tools from Asciidoctor project were used to produce the content for this book. Print formatting was done with Apache FOP from the Apache Graphics Project. Text is typeset in the Noto Serif font from Google. Code samples use a combination of Noto Mono from Google and M+ 1p from the M+ Fonts Project.

To Aaee

Miss you terribly

Table of Contents

List of Figures

List of Tables

Preface

...or why I wrote this book.

This book, as the more perceptive readers would have deduced from the title, is about the Tcl programming language.

About Tcl

Tcl was one of the first "dynamic" languages to become popular, seeing widespread use beginning with the early 90's. Along with its accompanying graphical programming toolkit Tk, the language was influential enough for John Ousterhout, its inventor, to be conferred the ACM Software System Award in 1998. Since that time, Tcl has found its way into every application category you can imagine [1] and Tcl deployments run the gamut from embedded devices like routers to Internet facing, galaxy class servers to distributed back-end infrastructure.

Yet, despite its wide use and adoption, Tcl has not gained the notoriety of the newer languages that have sprung up in the past few years. In large part this is because the Tcl community as a whole has never been particularly interested in evangelizing the language. As a result, people's perception of Tcl has remained stuck in the last century in terms of features and capabilities.

This book hopes to remedy that by providing comprehensive coverage of the current version, 8.6 as of writing, of Tcl. It aims to be a tutorial, programming guide and reference for the language and its runtime environment.

About the book

I have tried to cover every single feature of the Tcl programming language and core libraries in the book. However, space limitations do not permit inclusion of the myriad third-party libraries and extensions to Tcl. Important and commonly used ones are touched upon briefly so you are made aware of their existence but they are not described in detail. Even the graphical toolkit Tk, which is associated so strongly with Tcl that most people refer to the two in conjunction as Tcl/Tk, is not included. It would double the size of the book by itself. For the same reason, the book also does not discuss the C programming interface to Tcl although ease of interfacing to C is one of the strengths of the language.

About me

> Give me a place to stand and I'll move the earth.
>
> — Archimedes *famous Greek mathematician*

> Give me Tcl and I'll ship on time.
>
> — Me *famous Tcl author*

Authoring is hard work. Now a fainéant I anient [2] so I don't mind hard work. But it is **tedious** hard work (Oh Lord, indexing!) which is a whole different kettle of fish. And there is not much money to be made in technical books, it being difficult to insert violence and gratuitous sex into the prose to sell a few more copies. So what was my motivation in writing this book? Likely the same as most authors of technical books — when you believe in a language or technology, you have an urge, a compulsion, to spread the word.

My association with Tcl includes intermittent stretches over a couple of decades. I have run engineering in start-ups where significant components of the product were developed in Tcl. I have also authored several open source extensions and libraries for Tcl. In that time I grew to be increasingly enamoured of the language for one primary reason — productivity. From the perspective of an individual programmer, this productivity results from Tcl's malleability and rich feature set that facilitate a number of different programming styles and architectural patterns. From a management perspective, Tcl is stable, portable and versatile enough to be of benefit throughout a product's life-cycle — from development, test, deployment to field support.

[1] See http://wiki.tcl.tk/37889.
[2] With apologies to Ogden Nash

My hope is that this book will in some small way help popularize Tcl — beat the drum, so to speak, with regard to its simplicity and power.

Acknowledgements

First up is of course Professor John Ousterhout without whom there would be no Tcl. Beyond that, I thank en masse

- the members of the Tcl Core Team (TCT) that now directs development and future direction of the language
- the contributors to the Tcler's Wiki[3], which contains a trove of illustrative examples and expository material about Tcl, and the Tcl Chat room where much illuminating discussion takes place
- the many individuals who are collectively responsible for the libraries and tools that complete the Tcl ecosystem.

I would also like to thank Arjen Markus and Doran Moppert who reviewed portions of the book with patience and an eagle eye.

In terms of the book itself, a shout out to the Asciidoctor[4] team whose open source publishing toolchain greatly eased the actual process of writing and producing both electronic and print formats. Outstanding software.

Contacting me

You may contact me via my website http://www.magicsplat.com which also happens to be the home of my Tcl-related blog, articles and software. Alternatively, you may reach me through the Tcler's Wiki or the comp.lang.tcl newsgroup, both of which I follow regularly.

Ashok P. Nadkarni
Bengaluru, May 2017

[3] http://wiki.tcl.tk
[4] http://www.asciidoctor.org