

# TCL

Good, Bad and Ugly

# TCL or "Tool Command Language"

- Simple Multi-Paradigm Language
- BSD 3 Clause Licence
- Runs on many platforms
- Easy to learn
- Simple C interface

# History

- Originally developed as a reusable command and configuration language for CAD tools
- Designed to be linked into programs easily
- Born on 1988
- Originally Written by John Ousterhout

# The Good

# Simple to Learn

- Good editor support, Vim and Emacs
- Commands look like: Noun Verb Stuff
- Helpful error messages
- Magic is frowned upon
- Great for a part-time programmer
- Stable over time
- Great community and resources online, start at <http://wiki.tcl.tk>
- 12 rules to understand syntax

# 12 Rules of Tcl, man n Tcl

- Commands
- Evaluation
- Words
- Double Quotes
- Argument Expansion
- Braces
- Command Substitution
- Variable Substitution
- Backslash Substitution
- Comments
- Order Substitution
- Substitution of word boundaries

# Hello World

What explanation would be complete with out hello world:

```
puts "hello world"
```

- line terminator is '\n' , ';' is optional
- puts appends '\n' by default, can be suppressed

# Interactive tclsh

- commands not recognized by tcl will be passed through to the shell, can call man etc.

```
gandalf% tclsh8.5
```

```
% pwd
```

```
/home/marc
```

```
%
```

# Error Messages

- It is quicker to generally invoke an error then to look it up
- Its like Cisco IOS where it walks you through the command syntax
-

# Error as docs demo

- Start with "the GRAND puBA stuff"
- End with "The grand puba stuff"
- Remember Noun Verb stuff , stuff may be optional
- Using string command for example

# String example

% string ?

unknown or ambiguous subcommand "?": must be bytelength, compare, equal, first, index, is, last, length, map, match, range, repeat, replace, reverse, tolower, totitle, toupper, trim, trimleft, trimright, wordend, or wordstart

# String example pt 2

```
% string totitle
```

```
wrong # args: should be "string totitle string ?  
first? ?last?"
```

```
%
```

```
% string totitle "the GRAND puBA stuff"
```

```
The grand puba stuff
```

# Stable over time

- Old TCL code generally works
- TEA or "TCL Extension Architecture" means that old C code will generally work or work with minor tweaks
- Stability is valued by the community
- long term community, there has been a tcl conference for the past 18 years
- integrated unit testing framework that tests the language and all tests ship with the code.

# Extensions

- TCLLIB
- TkCon
- SNIT, pure tcl object system
- expect
- metakit, embedded nonsql db
- sqlite
- postgres
- mysql

- Tnm/scotty, snmp
- tls
- BLT

If you look you will probably find what you want

# Cisco

Tcl is embedded in your Cisco devices if they run IOS. There is a book on the subject "TcL Scripting for Cisco IOS", <http://www.ciscopress.com/bookstore/product.asp?isbn=1587059452>

Cisco CNR also uses tcl

# TclKit/StartKit

- tclkit and startkits are used to create deployable binaries from tcl scripts.
- they can embed compiled object code
- works on unix, linux, macos and windows
- the startkit can dynamically detect which os you are on and load the correct compiled code, with a few lines of code

# Start kit micro howto

- make a dummy file, bobo.tcl and have it print "hello world"
- run `./tclkit sdx.kit qwrap bobo.tcl` , you now have a start kit
- run `./tclkit bobo.kit` , see "hello world" print its alive
- run `./tclkit sdx.kit unwrap bobo.kit` , you now have a properly laid out virtual file system
- Hack and make app
- run `./tclkit sdx.kit wrap bobo.kit`
- run your app

**start kit demo**

**the Bad**

# Where the pain is

- Still no UDP in base, their are extensions though
- no parse step, very dynamic language so it defies static analysis.
- event loop can block
- white space sensitive in a different way then most scripting languages.
- you can override any command, this includes for loops and the like.

# Resources

- <http://wiki.tcl.tk>
- [comp.lang.tcl](http://comp.lang.tcl)
- <http://www.tcl.tk>
- IRC details here: <http://wiki.tcl.tk/1178>

# goodies in tcllib that I like

Highlight/sections:  
programming tools  
mathematics  
data structures  
text processing  
networking  
cgi programming

Specifics:  
ldap/ldapx  
smtp/mime  
report/csv/struct::  
matrix  
tcllib\_ip  
textutils  
cmdline  
pki

**Thanks and any  
questions?**

Marc Spitzer  
mspitzer@gmail.com