COMMANDS

after - Execute a command after a time delay

after **ms**
after **ms** ?script script script ...?
after **cancel** id
after **cancel** script script script ...
after **idle** ?script script script ...?
after **info** ?id?

append varName ?value value value ...? - Append to variable

auto_execok cmd
auto_import pattern
auto_load cmd
auto_mkindex dir pattern pattern ...
auto_mkindex_old dir pattern pattern ...
auto_qualify command namespace
auto_reset

break - Abort looping command

binary - Insert and extract fields from binary strings

binary **format** formatString ?arg arg ...?
binary **scan** string formatString ?varName varName ...?
format: a A b B h H c s S i I w W f d x X @

concat ?arg arg ...? - Join lists together

catch script ?varName? - Evaluate script and trap exceptional returns

continue - Skip to the next iteration of a loop

error message ?info? ?code? - Generate an error

eval arg ?arg ...? - Evaluate a Tcl script

for start test next body - ``For'' loop

foreach varlist1 list1 ?varlist2 list2 ...? body
- Iterate over all elements in one or more lists

format formatString ?arg arg ...? - Format a string in the style of sprintf

-
+
space
0
#
d
u
i
o
x or X
c
s
f
e or e
g or G
%

incr varName ?increment? - Increment the value of a variable

join list ?joinString? - Create a string by joining together list elements

lappend varName ?value value value ...? - Append list elements onto a variable

lindex list ?index...? - Retrieve an element from a list

linsert list index element ?element element ...? - Insert elements into a list

list ?arg arg ...? - Create a list

llength list - Count the number of elements in a list

lrange list first last - Return one or more adjacent elements from a list

lreplace list first last ?element element ...?
- Replace elements in a list with new elements

lsearch ?options? list pattern - See if a list contains a particular element

-**all**
-**ascii**
-**decreasing**
-**dictionary**
-**exact**
-**glob**
-**increasing**
-**inline**
-**integer**
-**not**
-**real**
-**regexp**
-**sorted**
-**start** index

lset varName ?index...? newValue - Change an element in a list

lsort ?options? list - Sort the elements of a list

-**ascii**
-**dictionary**
-**integer**
-**real**
-**command** command
-**increasing**
-**decreasing**
-**index** index
-**unique**

regexp ?switches? exp string ?matchVar? ?subMatchVar subMatchVar ...?
- Match a regular expression against a string

-**about**
-**expanded**
-**indices**
-**line**
-**linestop**
-**lineanchor**
-**nocase**
-**all**
-**inline**
-**start** index
- -

regsub ?switches? exp string subSpec ?varName?
- Perform substitutions based on regular expression pattern matching

-**all**
-**expanded**
-**line**
-**linestop**
-**lineanchor**
-**nocase**
-**start** index
- -

Syntax of Tcl regular expressions.

regular expresion syntax

*
+
?
{m}
{m,}
{m,n}
*? +? ??
{m}? {m,}?
{m,n}?
(re)
(?:re)
()
(?:)
[chars]
.
\k
\c
{
x
^
$
(?=re)
(?!re)

bracket expresions escapes

\a
\b
\B
\cX
\e
\f
\n
\r
\t
\uwxyz
\Ustuvwxyz
\v
\xhhh
\0
\xy
\xyz
\d
\s
\w
\D
\S
\W
\A
\m
\M
\y
\Y
\Z
\m
\mnn

metasyntax: b c e i m n p q s t w **x**

scan string format ?varName varName ...?
- Parse string using conversion specifiers in the style of sscanf

d
o
x
u
i
c
s
e or f or g
[chars]
[^chars]

n

string - Manipulate strings

string **bytelength** string
string **compare** ?-**nocase**? ?-**length** int?
string1 string2
string **equal** ?-**nocase**? ?-**length** int?
string1 string2
string **first** sample string ?startIndex?
string **index** string charIndex
charIndex: integer **end** **end**-integer
string **is** class ?-**strict**? ?-**failindex** varname?
string
class: alnum alpha ascii boolean
control digit double false graph
integer lower print punct space
true upper wordchar xdigit
string **last** sample string ?lastIndex?
string **length** string
string **map** ?-**nocase**? charMap string
string **match** ?-**nocase**? pattern string
pattern: * ? [chars] \x
string **range** string first last
string **repeat** string count
string **replace** string first last ?newstring?
string **tolower** string ?first? ?last?
string **totitle** string ?first? ?last?
string **toupper** string ?first? ?last?
string **trim** string ?chars?
string **trimleft** string ?chars?
string **trimright** string ?chars?
string **wordend** string charIndex
string **wordstart** string charIndex

subst ?-**nobackslashes**? ?-**nocommands**? ?-**novariables**? string
- Perform backslash, command, and variable substitutions

split string ?splitChars? - Split a string into a proper Tcl list

if expr1 ?**then**? body1 **elseif** expr2 ?**then**? body2 **elseif** ... ?**else**? ?bodyN?
- Execute scripts conditionally

return ?-**code** code? ?-**errorinfo** info? ?-**errorcode** code? ?string?
- Return from a procedure

code: ok error return break continue value

switch ?options? string pattern body ?pattern body ...?
switch ?options? string { pattern body ?pattern body ...? }
- Evaluate one of several scripts, depending on a given value

options: -**exect** -**glob** -**regexp** - -

update ?**idletasks**? - Process pending events and idle callbacks

uplevel ?level? arg ?arg ...? - Execute a script in a different stack frame

level is relative integer number, if preceeded by hash # then is absolute

vwait varName - Process events until a variable is written

while test body - Execute script repeatedly as long as a condition is met

array option arrayName ?arg arg ...? - Manipulate array variables

array **anymore** arrayName searchId
array **donesearch** arrayName searchId
array **exists** arrayName
array **get** arrayName ?pattern?
array **names** arrayName ?mode?
?pattern?
array **nextelement** arrayName searchId
array **set** arrayName list
array **size** arrayName
array **startsearch** arrayName
array **statistics** arrayName
array **unset** arrayName ?pattern?

global varname ?varname ...? - Access global variables

namespace ?option? ?arg ...? - Create and manipulate new contexts

namespace **children** ?namespace? ?pattern?
namespace **code** script
namespace **current**
namespace **delete** ?namespace namespace ...?
namespace **eval** namespace arg ?arg ...?
namespace **exists** namespace
namespace **export** ?-**clear**? ?pattern pattern ...?
namespace **forget** ?pattern pattern ...?
namespace **import** ?-**force**? ?pattern pattern ...?
namespace **inscope** namespace script ?arg ...?
namespace **origin** command
namespace **parent** ?namespace?
namespace **qualifiers** string
namespace **tail** string
namespace **which** ?-**command**?
?-**variable**? name

proc name args body - Create a Tcl procedure

rename oldName newName - Rename or delete a command

set varName ?value? - Read and write variables

trace option ?arg arg ...? - Monitor variable and commands

trace **add** (**command**|**execution**|**variable**) name ops ?args?
command: rename delete
execution: enter leave enterstep leavestep
variable: array read write unset
trace **remove** (**command**|**execution**|**variable**) name opList command
trace **info** (**command**|**execution**|**variable**) name
trace **variable** name ops command
trace **vdelete** name ops command
trace **vinfo** name

unset ?-**nocomplain**? ?--? ?name name name ...? - Delete variables

upvar ?level? otherVar myVar ?otherVar myVar ...?
- Create link to variable in a different stack frame

variable ?name value...? name ?value?
- create and initialize a namespace variable

close channelId - Close an open channel.

eof channelId - Check for end of file condition on channel

fblocked channelId
- Test whether the last input operation exhausted all available input

fconfigure channelId ?name? ?value? ?name value ...?
- Set and get options on a channel

-**blocking** boolean
-**buffering** newValue
-**buffersize** newSize
-**encoding** name
-**eofchar** char
-**eofchar** {inChar outChar}
-**translation** mode
-**translation** {inMode outMode}
auto binary cr crlf lf

fcopy inchan outchan ?-**size** size? ?-**command** callback?
- Copy data from one channel to another.

file option name ?arg arg ...? - Manipulate file names and attributes

file **atime** name ?time?
file **attributes** name ?option?
?value option value...?
file **channels** ?pattern?
file **copy** ?-**force**? ?- -? source target
file **copy** ?-**force**? ?- -? source
?source ...? targetDir
file **delete** ?-**force**? ?- -? pathname
?pathname ... ?
file **dirname** name
file **executable** name
file **exists** name
file **extension** name
file **isdirectory** name
file **isfile** name
file **join** name ?name ...?
file **link** ?-**symbolic**|**hard**? linkName ?target?
file **lstat** name varName
file **mkdir** dir ?dir ...?
file **mtime** name ?time?
file **nativename** name
file **normalize** name
file **owned** name
file **pathtype** name
file **readable** name
file **readlink** name
file **rename** ?-**force**? ?- -? source target
file **rename** ?-**force**? ?- -? source
?source ...? targetDir
file **rootname** name
file **separator** ?name?
file **size** name
file **split** name
file **stat** name varName
file **system** name
file **tail** name
file **type** name
file **volumes**
file **writable** name

fileevent channelId **readable**|**writable** ?script?
- Execute a script when a channel becomes readable or writable

flush channelId - Flush buffered output for a channel

gets channelId ?varName? - Read a line from a channel

open fileName ?access ?permissions??
- Open a file-based or command pipeline channel

permisions: r r+ w w+ a a+ RDONLY WRONLY RDWR APPEND CREAT EXCL NOCTTY
NONBLOCK TRUNC

puts ?-**nonewline**? ?channelId? string - Write to a channel

read ?-**nonewline**? channelId ?numChars? - Read from a channel

seek channelId **start**|**current**|**end** ?origin?
- Change the access position for an open channel

socket ?options? host port
socket -**server** command ?options? port
- Open a TCP network connection

tell channelId - Return current access position for an open channel

load fileName ?packageName? ?interp?
- Load machine code and initialize new commands

loadTk - Load Tk into a safe interpreter

package - Facilities for package loading and version control

package **forget** ?package package ...?
package **ifneeded** package version ?script?
package **names**
package **present** ?-**exact**? package ?version?
package **provide** package ?version?
package **require** ?-**exact**? package ?version?
package **unknown** ?command?
package **vcompare** version1 version2
package **versions** package
package **vsatisfies** version1 version2

pkg::create -**name** pkgName -**version** pkgVersion ?-**load** filespec? ... ?-**source** filespec? ...
- Construct an appropriate package ifneeded

pkg_mkIndex ?-**direct**? ?-**lazy**? ?-**load** pkgPat? ?-**verbose**? dir ?pattern pattern ...?
- Build an index for automatic loading of packages

source fileName - Evaluate a file or resource as a Tcl script (see mac opt.)

bgerror message - Command invoked to process background errors

history ?option? ?arg arg ...? - Manipulate the history list

history
history **add** command ?exec?
history **change** newValue ?event?
history **clear**
history **event** ?event?
history **info** ?count?
history **keep** ?count?
history **nextid**
history **redo** ?event?

info option ?arg arg ...? - Return information about the state of the Tcl interpreter

info **args** procname
info **body** procname
info **cmdcount**
info **commands** ?pattern?
info **complete** command
info **default** procname arg varname
info **exists** varName
info **functions** ?pattern?
info **globals** ?pattern?
info **hostname**
info **level** ?number?
info **library**
info **loaded** ?interp?
info **locals** ?pattern?
info **nameofexecutable**
info **patchlevel**
info **procs** ?pattern?
info **script** ?filename?
info **sharedlibextension**
info **tclversion**
info **vars** ?pattern?

interp option ?arg arg ...? - Create and manipulate Tcl interpreters

memory option ?arg arg ...? - Control Tcl memory debugging capabilities

unknown cmdName ?arg arg ...? -Handle attempts to use non-existent commands

encoding option ?arg arg ...? - Manipulate encodings

encoding **convertfrom** ?encoding? data
encoding **convertto** ?encoding? string
encoding **names**
encoding **system** ?encoding?

http * - Client-side implementation of the HTTP/1.0 protocol

msgcat - Tcl message catalog

::msgcat::mc src-string ?arg arg ...?
::msgcat::mcmax ?src-string src-string ...?
::msgcat::mclocale ?newLocale?
::msgcat::mcpreferences
::msgcat::mcload dirname
::msgcat::mcset locale src-string ?translate-string?
::msgcat::mcmset locale src-trans-list
::msgcat::mcunknown locale src-string

cd ?dirName? - Change working directory

clock format clockValue ?-**format** string? ?-**gmt** boolean?
%% %a %A %b %B %c %C %d %D %e %h %H %I %j %k %l %m %M %n %p %r %R %s %S %t %T %u %U %V %w %W %x %X %y %Y %Z

clock scan dateString ?-**base** clockVal? ?-**gmt** boolean?

clock seconds

exec ?switches? arg ?arg ...? - Invoke subprocess(es)

exit ?returnCode? - End the application

glob ?switches? pattern ?pattern ...? - Return names of files that match patterns
switches: -**directory** directory -**join** -**nocomplain** -**path** pathPrefix -**tails** -**types** typeList - -
types: b (block special file), c (character special file), d (directory), f (plain file), l (symbolic link), p (named pipe), s (socket), r, w, x, readonly, hidden

pid ?fileId? - Retrieve process id(s)

pwd - Return the current working directory

time script ?count? - Time the execution of a script

dde * - Execute a Dynamic Data Exchange command

registry - Manipulate the Windows registry

registry **delete** keyName ?valueName?
registry **get** keyName valueName
registry **keys** keyName ?pattern?
registry **set** keyName ?valueName data ?type??
registry **type** keyName valueName
registry **values** keyName ?pattern?
type: binary none sz expand_sz dword dword_big_endian link multi_sz resource_list

resource * - Manipulate Macintosh resources

tcl_findLibrary basename version patch initScript enVarName varName
parray arrayName
tcl_endOfWord str start
tcl_startOfNextWord str start
tcl_startOfPreviousWord str start
tcl_wordBreakAfter str start
tcl_wordBreakBefore str start

OTHER

filename

mac		
:	:MyDir:MyFile	/MyDisk/MyFile
MyFile	::MyFile	../MyFile
MyDisk:MyFile	:::MyFile	
unix		
/	.	foo/bar
/etc/passwd	foo	../foo
windows		
\\Host\share\file	c:/foo	\foo
c:foo	foo\bar	\\foo
default chanell names		
stdin	**stdout**	**strerr**

expresions

operators		
- + ~ !	== !=	&&
* / %	eq ne	||
+ -	&	x**?**y**:**z
<< >>	^	
< > <= >=	|	
math functions		
abs(arg)	**exp**(arg)	**round**(arg)
acos(arg)	**floor**(arg)	**sin**(arg)
asin(arg)	**fmod**(x, y)	**sinh**(arg)
atan(arg)	**hypot**(x, y)	**sqrt**(arg)
atan2(y, x)	**int**(arg)	**srand**(arg)
ceil(arg)	**log**(arg)	**tan**(arg)
cos(arg)	**log10**(arg)	**tanh**(arg)
cosh(arg)	**pow**(x, y)	**wide**(arg)
double(arg)	**rand**()	

Tcl 8.4.6

http://www.tcl.tk/

quickcard ver. 0.3 by michael.heca@email.cz

GENERAL

<**newline**> or „**;**“	Command separation
“evaluated string**“**	Double qoutet strings are evaluated.
{fixed string/list**}**	Braced strings/list are not evaluated.
[command arguments...**]**	String in bracked are replaced by command evaluation result.
# comment	Comments are preceded with hash, only one line, braces in comment must bee balanced

backslash substitution

a	Audible alert (bell) (0x7).
b	Backspace (0x8).
f	Form feed (0xc).
n	Newline (0xa).
r	Carriage-return (0xd).
t	Tab (0x9).
v	Vertical tab (0xb).
\\<**newline**>**whiteSpace**	Newline substitution with word separator.
\	Backslash (``\").
\ooo	Eight-bit octal value for the Unicode character that will be inserted.
\xhh	Eight-bit hexadecimal value for the Unicode character
\uhhhh	Sixteen-bit hexadecimal value for the Unicode character

variable substitution

$name	**${**name**}**	**$**name**(**index**)**

SYSTEM VARIABLES

auto_execs auto_index auto_noexec auto_noload auto_path

argv – list of program arguments

argc – count of program arguments

argv0 – program file name

tcl_interactive – 1 if run interactively

env – enviroment variable array

LOGIN	DESK_FOLDER	TCL_LIBRARY
USER	EXT_FOLDER	TCLLIBPATH
SYS_FOLDER	PREF_FOLDER	TRASH_FOLDER
APPLE_M_FOLDER	PRINT_MON_FOLDER	START_UP_FOLDER
CP_FOLDER	SHARED_TRASH_FOLDER	HOME

errorCode – list holding last error information

ARITH code msg	CHILDSUSP pid sigName msg
CHILDKILLED pid sigName msg	NONE
CHILDSTATUS pid code	POSIX errName msg

errorInfo – string information about last error

tcl_platform – platform information array

byteOrder	os	threaded
debug	osVersion	user
machine	platform	wordSize

tcl_library
tcl_nonwordchars
tcl_patchLevel
tcl_pkgPath
tcl_precision
tcl_rcFileName
tcl_rcRsrcName
tcl_traceCompile
tcl_traceExec
tcl_version
tcl_wordchars
unknown_pending