

C Runtime In Tcl v3

Andreas Kupries
ActiveState Software Inc.
© 2011

18th Annual Tcl Conference
Comfort Suites, Manassas, VI
Oct 24 – Oct 28 2011

CriTcl v3

- Background
- New Features
- Future
- Miscellanea

CriTcl v3

- **Background**
- New Features
- Future
- Miscellanea

CriTcl v3 - Background

- CRIMP
 - Based on CriTcl.
 - Required support for Tcl stubs
 - Separation of the Tk-dependent parts from the overall system, using the same data structures

CriTcl v3

- Background
- **New Features**
- Future
- Miscellanea

CriTcl v3 – New Features

- Export and Import of stubs tables
- Generation of TEA package from the sources
- Declaration and use of user-specified configuration settings
- Declaration and generation of TEApot meta data.
- Better control over package loading.

CriTcl v3 – Stubs

- Declare and export an API
 - critcl::api header cstack.h
 - Header to export with types, macros, ...
 - critcl::api function CSTACK cstack_new ...
 - Function to export
- Critcl generates the necessary files.
 - cstack/cstackDecls.h
 - cstack/cstackStubLib.h
 - cstack/cstack.decls

CriTcl v3 – Stubs

- Use an API
 - `critcl::api import cstack 1`
- Critcl locates the generated headers and inserts the necessary includes into generated code.
 - In companion files (.c, .h) necessary to include manually.
- Programmatic access to the contents of .decls, if present.
 - Automatic generation of a Tcl interface from the API declarations.

CriTcl v3 – TEA

- “critcl -tea package.tcl”
 - Scans the sources for companion files and collects them.
 - Generates configure(.in), Makefile.in
 - Result is a directory hierarchy which can be build via
 - configure ; make ; make install

CriTcl v3 – User Config

- **examples/zlibwrapper**
 - Declare
 - `critcl::userconfig define mode {...} system system-static local`
 - Call
 - `critcl -with-mode ... zlib.tcl`
 - Use
 - `switch – [critcl::userconfig query mode] ...`
 - Controls linking of the zlib wrapper
 - Against system `-lz`, dynamic or static (`system*`)
 - Or use local copy of zlib sources (`local`)

CriTcl v3 – Meta data

- Commands
 - critcl::license (extended from v2)
 - critcl::summary
 - critcl::description
 - critcl::subject
 - critcl::meta key value (generic)
- Plus data collected from existing commands
 - Provided & required packages, apis, platform, timestamps, ...
- Critcl generates and places teapot.txt

CriTcl v3 – Loading

- Compile & run, by default **lazy**
 - `auto_index` → Compile and load on demand
 - Introspection via `[info commands]` fails → Want immediate built & loading
- v2 has `[critcl::failed]`.
 - Use case: Check for build errors.
 - Forces build, but not load.
- **New** “`critcl::load`” command forces both.

CriTcl v3

- Background
- New Features
- **Future**
- Miscellanea

CriTcl v3 - Future

- Currently not working on anything
 - I have what I wanted and needed 😊

CriTcl v3

- Background
- New Features
- Future
- **Miscellanea**

CriTcl v3 - Miscellanea

- The **svn** repository at equi4.com is **deprecated**
- New location @ **github**
 - Official repository, by Jean-Claude Wippler.
 - <https://jcw.github.com/critcl>
 - <https://github.com/jcw/critcl>
 - Backup repository, origin of the v3 work.
 - <https://andreas-kupries.github.com/critcl>
 - <https://github.com/andreas-kupries/critcl>