Explorations in GUI programming with CHICKEN

Vasilij Schneidermann

September 2016

Outline

1 Intro

2 Basics

3 Existing Eggs

4 New Eggs

5 Outro

Section 1

Intro

About

- Vasilij Schneidermann, 24
- Information systems student
- Working at bevuta IT, Cologne
- v.schneidermann@gmail.com
- https://github.com/wasamasa
- http://emacshorrors.com/
- http://emacsninja.com/

How it came to be

- Thesis about CS and or economics
- Difficulties finding a topic
- Employer suggested CHICKEN
- -> CHICKEN's GUI programming support

General approach

- Looking at existing GUI libraries
- Understanding how GUIs work
- Reading (mostly useless) papers
- Surveying existing eggs
- Finding approach for developing new eggs
- Evaluating new eggs
- Recommendations for beginners

Section 2

Basics

Approaches

- Pure Scheme library (Racket, JazzScheme, qobischeme-ui)
- Porting existing Lisp library (McCLIM, guile-gnome, stklos-gtk-base)
- Graphical shell (Tcl/tk, ezd, Rebol, Red)
- Other approaches
  - WebKit (WebKitGTK+, QtWebKit, QtWebEngine, NW.js, Electron)
  - GObject Introspection (sbank, cl-gir, seed, Gjs)
- Wrapping foreign code
  - Java (AWT, Swing, SWT, JAPI)
  - C++ (Qt, wxWidgets, JUCE, FLTK, CEGUI, neogfx, dear imgui)
  - C (libsx, KiWi, mrg, nuklear, blendish/OUI, mx and Clutter, libui, Agar, GTK, Tcl/Tk, IUP)

Programming paradigms

- Procedural (good fit, a tad inelegant)
- OO (bad fit)
- Functional (unknown, FRP, Elm architecture)

Retained vs. Immediate Mode

- Retained (traditional, callbacks/event handlers)
- Immediate (dynamic, simple, less efficient)

Native vs. non-native widgets

- Native (pretty, feels right, hard to integrate, "heavy-weight")
- Non-native (flexible, portable, simple, ugly)

GUI builders

- Necessary for huge and complicated GUI toolkits (Qt)
- Can be avoided in Scheme (REPL-driven development)
- SXML/ASCII interface

Section 3

Existing Eggs

tk and PS/Tk

- Pure Scheme interfaces to Tcl/Tk, everything accessible
- Not native (somewhat themable), layout managers
- Very simple, good for beginners
- tk Egg has weird license, PS/Tk is marked obsolete and has no docs
- http://www.slideshare.net/r1chardj0n3s/tkinter-does-not-suck
- Alternative: Use pure Tcl, communicate with Scheme process (feathers)
- http://philip.greenspun.com/tcl/index.adp
- http://antirez.com/articoli/tclmisunderstood.html

bb

- Abstraction over FLTK
- SXML/ASCII interface
- Absolute positioning necessary, not native (ugly)
- Small subset of FLTK API, unclear how useful in practice
- Good docs!

iup

- C interface to IUP
- Native widgets for Win32 and GTK2/3
- Very nice API, good docs
- Hard to build, broken on anything newer than Ubuntu 15.04

qt-light

- Tiny subset of Qt4 (installation assumes it's default. . .)
- Native look, not quite native behavior
- You must use `designer`
- Unclear how useful in practice

webkit

- Not exactly a GUI toolkit. . .
- Walking security problem
- Style with CSS (flexbox), script with JS (SPOCK?)
- Injection of Scheme callbacks
- Spawn windows to `file:///...`
- Somewhat buggy (focus, finalizer warnings)

Section 4

New Eggs

kiwi

- Neat SDL2 toy, wrote a PKGBUILD
- Event handlers, non-native
- Very few widgets (5!)
- Theme by tileset
- Absolute positioning only
- Most API is accessible, completely documented
- Playground for binding strategies

nuklear

- Inspired by dear imgui, intended for game (debugging)
- Header library, very little dependencies
- Immediate, works against OpenGL-like backends
- Imperative API, no callbacks/event handlers
- Extremely dynamic, fits well to games
- Custom widgets are easy
- Bindings to example code only
- Docs had to be written from scratch

libui

- IUP-style project for Win32, Cocoa, GTK3
- Had to write my own PKGBUILD (again)
- Very immature, little docs
- API is acceptable
- SXML interface for less imperative code
- Event handlers

Section 5

Outro

Summary

- Tcl/Tk is surprisingly good
- WebKit can be useful for special usecases
- IUP is perfect in theory, libui can be a replacement for it
- nuklear is interesting

Research topics

- Bindings to bigger libraries (clutter, Agar, Qt via SMOKE)
- GObject wrappers (SPOCK for seed/Gjs?)
- Declarative interface for QtQuick/QML
- Graphical REPLs (ezd, REBOL/Red?)
- I'd love seeing more immediate mode GUIs, ideally in Scheme
- Functional GUI toolkits please!

Questions?