

Using Vim as User Interface for Your Applications

Jérôme Belleman

IT-CM-LCS

22 September 2017

The Challenge of Making Good User Interfaces

- User interfaces are sophisticated
- Often too much so
- And yet usually strangely limited
- Only few succeeded: Vim

The Challenge of Making Good User Interfaces

- User interfaces are sophisticated
- Often too much so
- And yet usually strangely limited
- Only few succeeded: Vim, Emacs, ...

A File Manager Inside Vim: netrw

```
% vim /
```

```
" =====  
" Netrw Directory Listing (netrw v145)  
" /  
" Sorted by name  
" Sort sequence: [\/]$, \<core%\(\.\d\+\)\=\>, \.h$, \.c$  
" Quick Help: <F1>:help -:go up dir D:delete R:rename  
" =====  
../  
./  
afs/  
bin/  
boot/  
cdrom/  
dev/  
etc/  
home/  
lib/  
lost+found/  
media/
```

Imagine. . .

- Not having to worry about the user interface – it's already there
- Letting users steer the app the way they want
- Allowing any key binding, any command, any script, . . .

Example: Dancing Club Management App

```
% cdc
```

```
-----  
Name                Signed Up?         Course  
-----  
Marion Albert       Yes               Salsa  
Silas Russell       Yes               Rock  
Mickey Mcdonald     Yes               Tango  
Althea Pope         No                Tango  
Kathie Patrick      Yes               Salsa  
Sterling Anderson  Yes               Salsa  
Dante Davis         Yes               Tango  
Lenora Hutchinson  Yes               Rock  
Harley Cohen        Yes               Salsa  
Fredric Barr        Yes               Tango  
Sophie Jefferson    Yes               Ballroom  
Olivia Norris       No                Ballroom  
Oscar Dalton        Yes               Rock
```

Bot

(Privacy notice: these are fake names – see the `rig` command)

Example: Dancing Club Management App

```
% cdc
```

```
Name                Signed Up?          Course
=====
Marion Albert       Yes                Salsa
Silas Russell       Yes                Rock
Mickey Mcdonald    Yes                Tango
Althea Pope         No                 Tango
Kathie Patrick     Yes                Salsa
Sterling Anderson  Yes                Salsa
Dante Davis        Yes                Tango
Lenora Hutchinson  Yes                Rock
Harley Cohen       Yes                Salsa
Fredric Barr       Yes                Tango
Sophie Jefferson   Yes                Ballroom
Olivia Norris      No                 Ballroom
Oscar Dalton       Yes                Rock

:sort / /
```

(Privacy notice: these are fake names – see the `rig` command)

Example: Dancing Club Management App

```
% cdc
```

```
-----  
Name                Signed Up?         Course  
-----  
Marion Albert       Yes                Salsa  
Sterling Anderson   Yes                Salsa  
Fredric Barr        Yes                Tango  
Harley Cohen        Yes                Salsa  
Oscar Dalton        Yes                Rock  
Dante Davis         Yes                Tango  
Lenora Hutchinson   Yes                Rock  
Sophie Jefferson    Yes                Ballroom  
Mickey Mcdonald     Yes                Tango  
Olivia Norris       No                 Ballroom  
Kathie Patrick      Yes                Salsa  
Althea Pope         No                 Tango  
Silas Russell       Yes                Rock
```

Bot

(Privacy notice: these are fake names – see the `rig` command)

How?

```
#!/usr/bin/env python

import subprocess

subprocess.call(['/usr/bin/vim', '-Scdc.vim'])
```

```
autocmd VimEnter <buffer> :python readdb()
autocmd VimLeave <buffer> :python writedb()
map <unique> <Leader>o :python showmember()<CR>

python << EOF
import vim

def readdb():
    for ...:
        vim.current.line = getnextmember()
EOF
```

Reference and Inspiration

- http://vimdoc.sourceforge.net/html/doc/if_pyth.html
- <http://vimdoc.sourceforge.net/html/doc/autocmd.html>
- <http://vimdoc.sourceforge.net/html/doc/map.html>