February 4, 2012

Ian Ward

Essential Details

High-level console user interface library for Python

LGPL

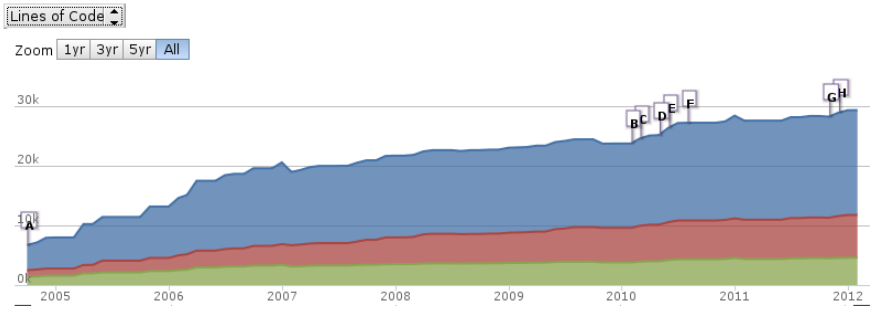
Unix-like OSes: Linux, OSX, (windows with Cygwin)

Urwid Development

http://www.ohloh.net/p/urwid/contributors

> 20 contributors over the course of the project

Mostly yours truly

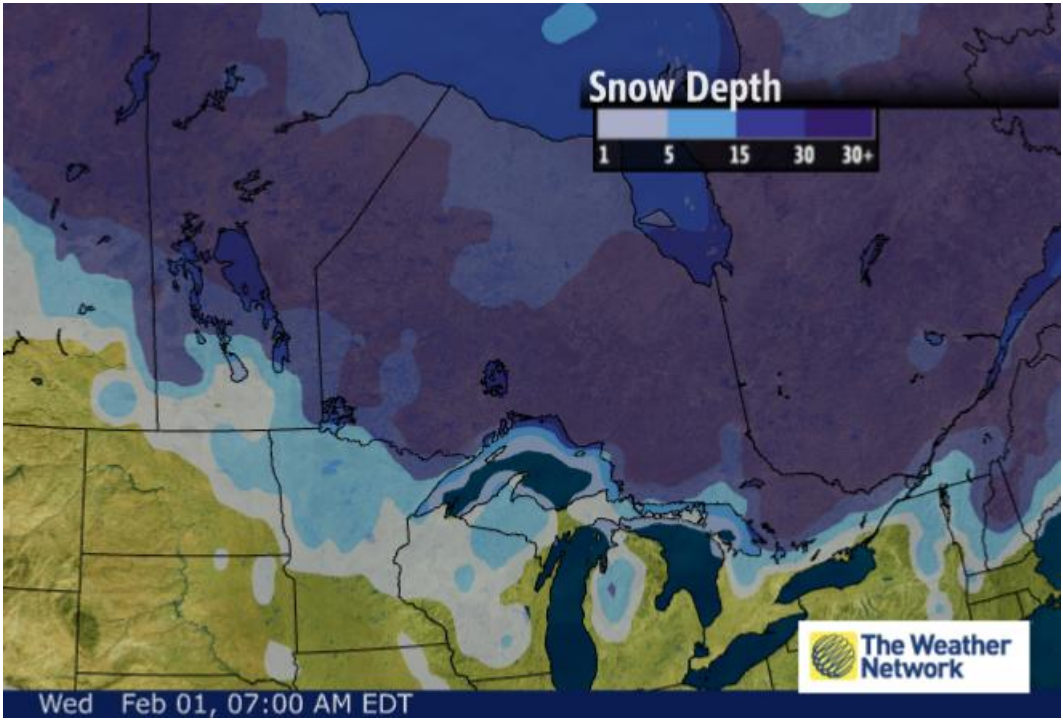


About Myself

- Independent software developer

  http://genericconsulting.ca/

  Django, PostgreSQL, Python, C, ASM
- located near Ottawa, Canada

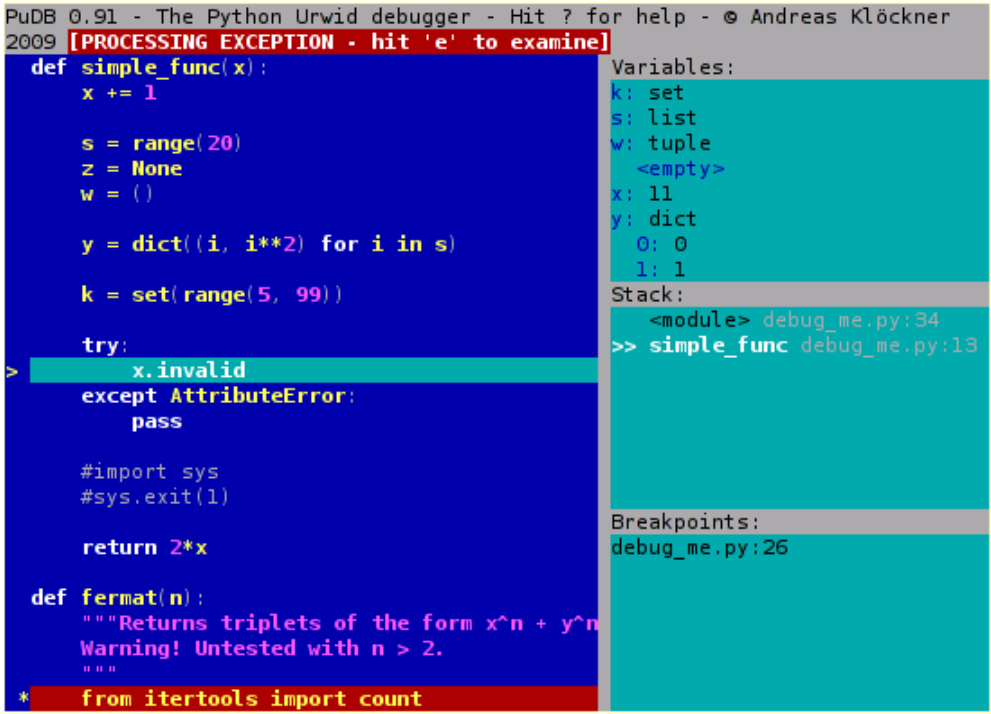

Snow Depth
1
5
15
30
30+
The Weather Network
Wed Feb 01, 07:00 AM EDT

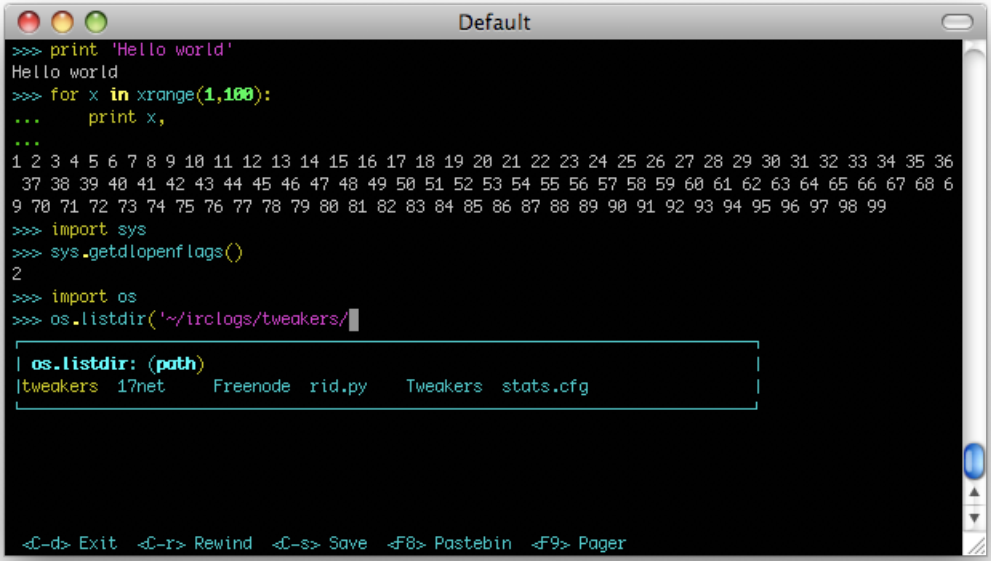
Urwid Users

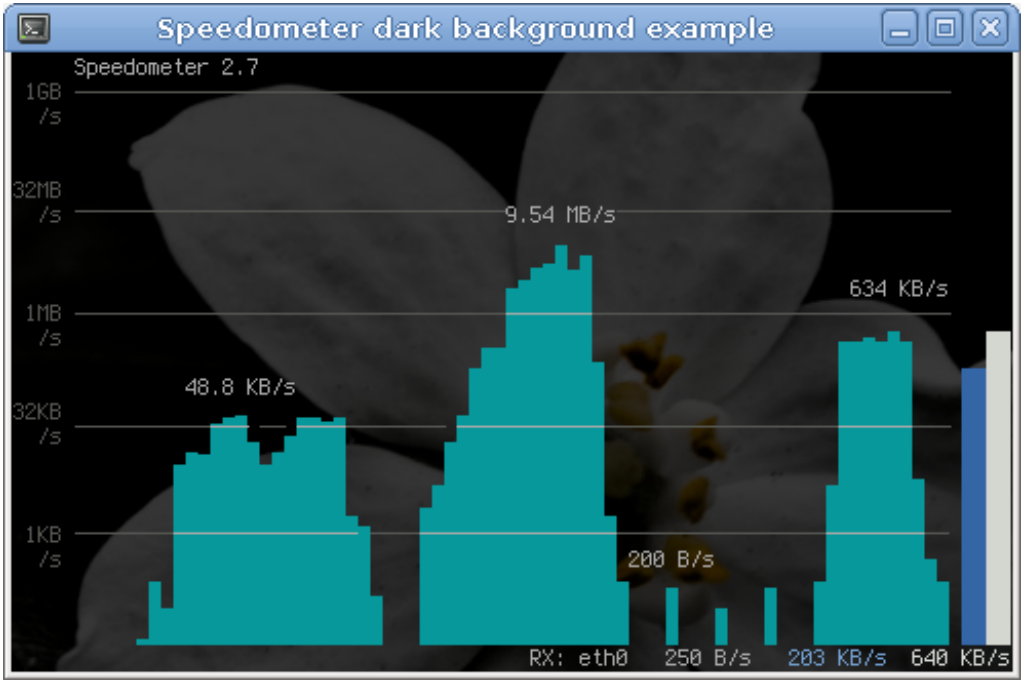
http://excess.org/urwid/wiki/ApplicationList

~ 20 projects listed, including

- bpython
- wicd
- pudb
- reportbug
- speedometer


PuDB 0.91 - The Python Urwid debugger - Hit ? for help - © Andreas Klöckner 2009
[PROCESSING EXCEPTION - hit 'e' to examine]
def simple_func(x):
x += 1
s = range(20)
z = None
w = ()
y = dict((i, i**2) for i in s)
k = set(range(5, 99))
try:
> x.invalid
except AttributeError:
pass
#import sys
#sys.exit(1)
return 2*x
def fermat(n):
"""Returns triplets of the form x^n + y^n
Warning! Untested with n > 2.
"""
* from itertools import count
Variables:
k: set
s: list
w: tuple
<empty>
x: 11
y: dict
0: 0
1: 1
Stack:
<module> debug_me.py:34
>> simple_func debug_me.py:13
Breakpoints:
debug_me.py:26


Default
>>> print 'Hello world'
Hello world
>>> for x in xrange(1,100):
...   print x,
...
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 6
9 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
>>> import sys
>>> sys.getdlopenflags()
2
>>> import os
>>> os.listdir('~/irclogs/tweakers/
os.listdir: (path)
|tweakers 17net Freenode rid.py Tweakers stats.cfg
<C-d> Exit <C-r> Rewind <C-s> Save <F8> Pastebin <F9> Pager


Speedometer dark background example
Speedometer 2.7
1GB /s
32MB /s
1MB /s
32KB /s
1KB /s
9.54 MB/s
634 KB/s
48.8 KB/s
200 B/s
RX: eth0 250 B/s 203 KB/s 640 KB/s

Good Console Interfaces

Helps you write apps like: vim, mc, w3m, rtorrent, aptitude, tmux

Doesn't (yet) help you make silly ones (turbovision)

Goals

- Use the complete display area efficiently
- Sparse UI that stays out of your way
- Work well over ssh, even with high latency
- No particular GUI required, text console is fine

Alternatives

curses

- comes with python
- very low level

newt/snack

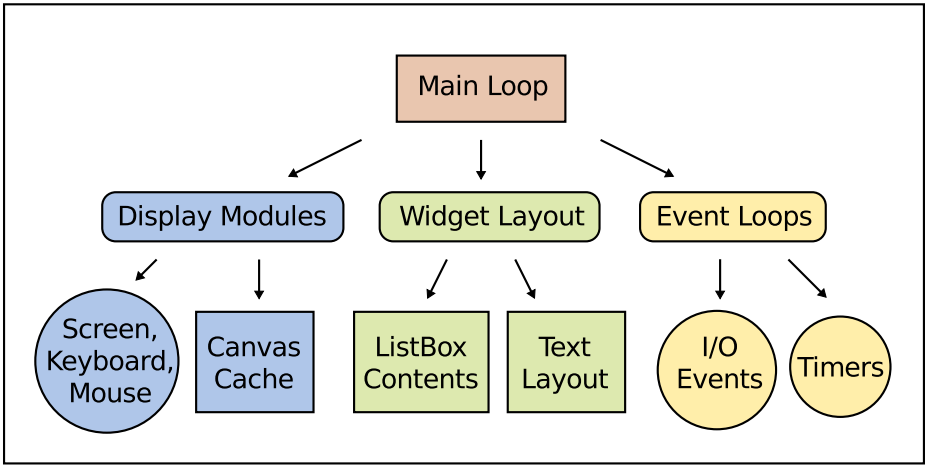
- some gui-like widgets

insults

- well integrated with twisted

Urwid

- > 7 years
- reached 1.0 in 2011
- supports Python 2.4-2.7, 3.2+
- good Unicode support
- friendly 256/88 colour support
- twisted/glib event loop support
- separation of concerns from the start


Main Loop
Display Modules
Widget Layout
Event Loops
Screen, Keyboard, Mouse
Canvas Cache
ListBox Contents
Text Layout
I/O Events
Timers

The Reason Urwid Exists

I wanted to write a point of sale system

- Text Layout
- Input Handling
- Lists and Scrolling

Text Layout is Hard

Wrapping, alignment, clipping, double-width and zero-width characters

http://excess.org/urwid/wiki/TextLayout

- new data stucture

  maps 1d positions <-> 2d coordinates
- caching
- custom text layouts
- editing + cursor positioning

Input Handling is Hard

- curses doesn't work very well

  many xterm key combinations unsupported

  termcap / terminfo can't be fixed
- do it ourself
- minimize delay on ESC
- support all the key combinations I want

Lists and Scrolling are Hard

ListBox

- cursor key and page up/down handling
- relative focus position
- selectable/unselectable widgets

List Walker

- focus tracking
- widget content generation

Unicode is Hard

- extend text layout
- raw_display

Event Loops are Hard

MainLoop

- pass input from user to widgets
- draw widgets on the screen
- throttle screen resizing
- do the fast stuff first

Event Loops

- wait on IO
- timers
- notice when idle

Penultimate Slide

Consider making a console UI for your next app, even just as a second option.

Contribute to OSS or build your own OSS project. Lots of work but it can be very satisfying.

Questions