Introducing Conky Bubbles

Simon Lees

sflees@suse.de / simon@simotek. net

https://github.com/simotek/conky-bubbles

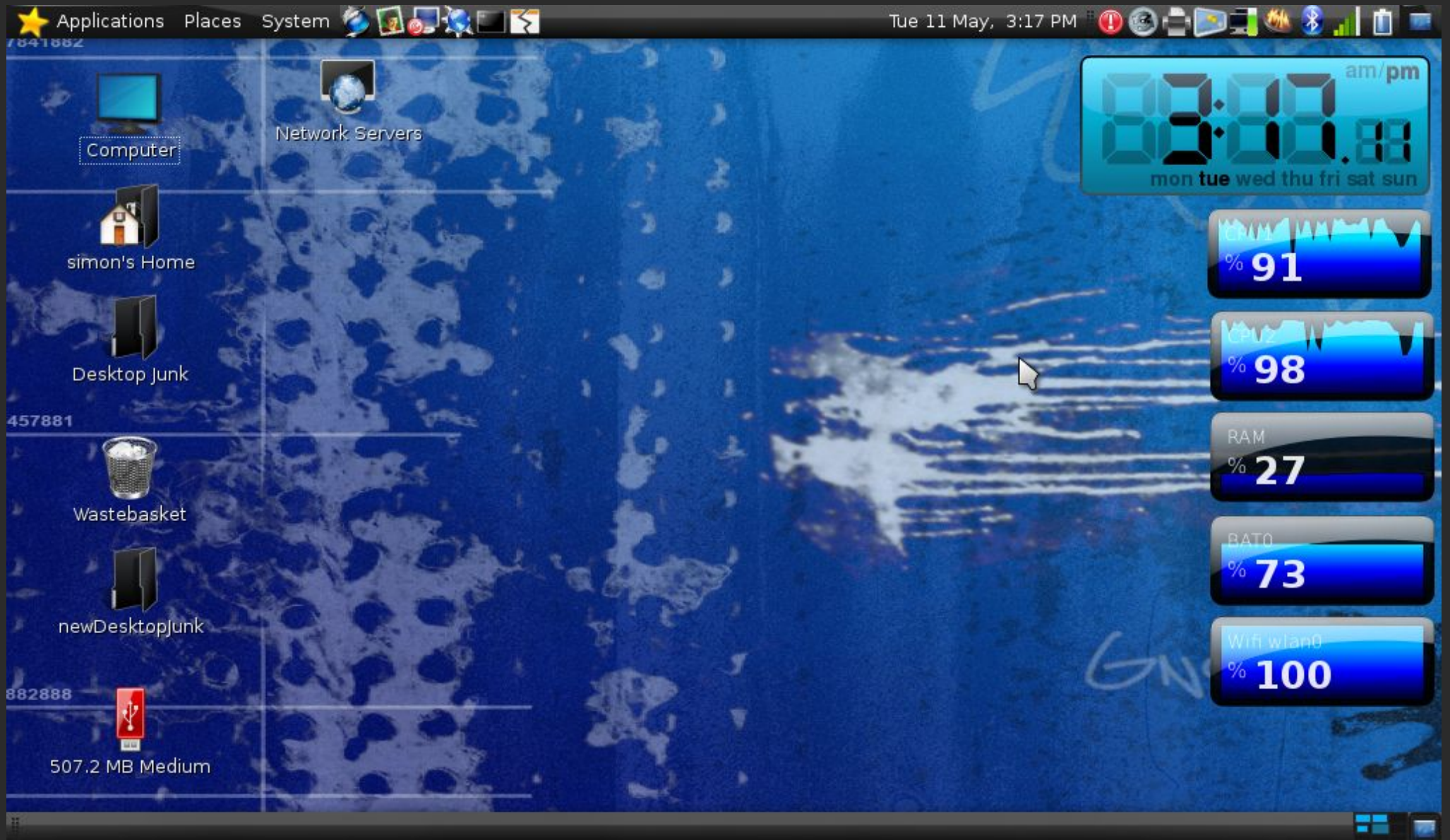
This talk

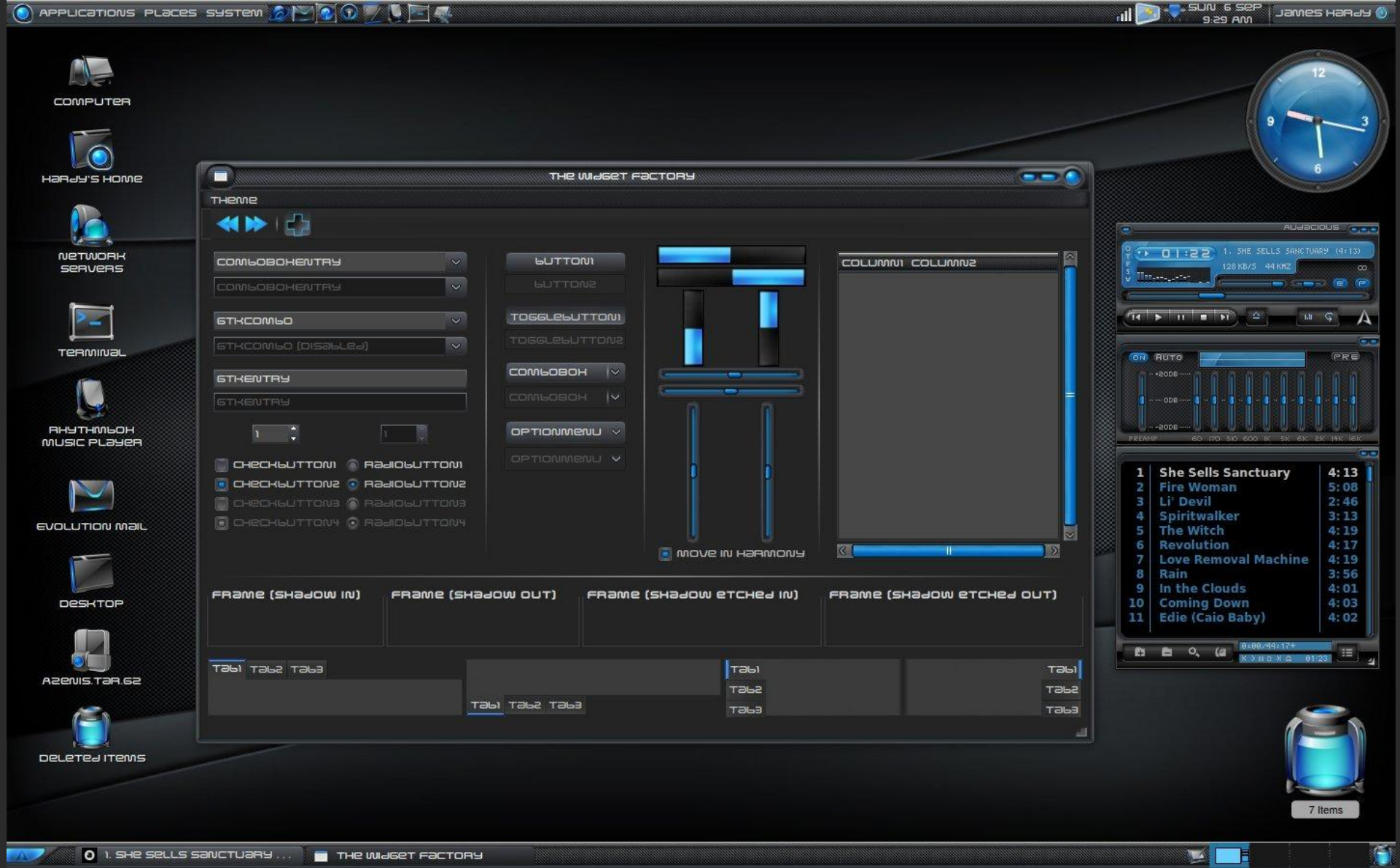
- What is Conky
- Why do we need this new project
- How to work with other existing open source projects
- Font rendering

About Me

- Contributing to open source for 15 years
- Packaging for Linux Distros for a similar amount of time
- Hobbies include UI Design and Customisation

Why?


Applications
Places
System
Tue 11 May, 3:17 PM
Computer
Network Servers
simon's Home
Desktop Junk
Wastebasket
newDesktopJunk
507.2 MB Medium
am/pm
mon tue wed thu fri sat sun
CPU1
% 91
CPU2
% 98
RAM
% 27
BAT0
% 73
Wifi wlan0
% 100

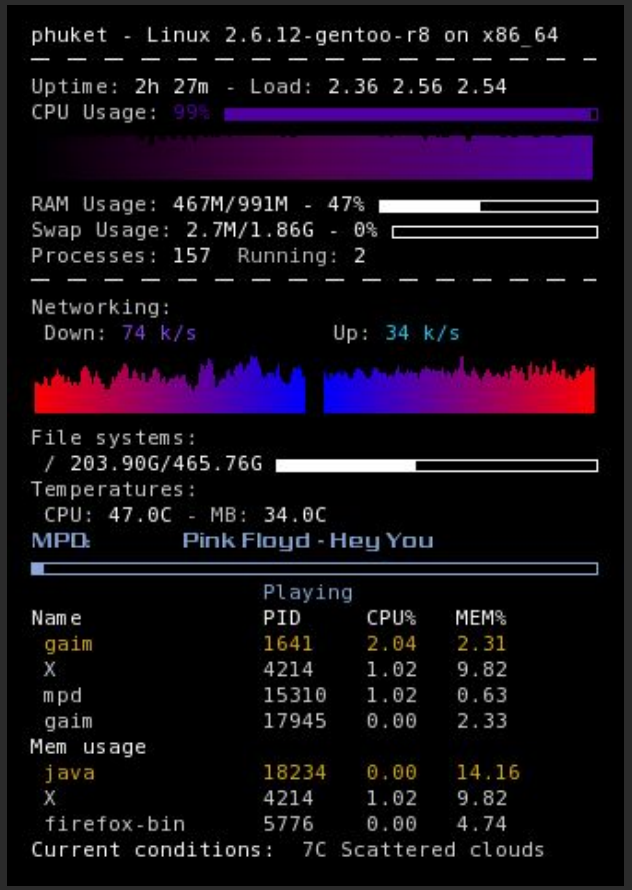

Applications Places System
Sun 6 Sep
9:29 AM
James Hardy
Computer
Hardy's Home
Network Servers
Terminal
Rhythmbox Music Player
Evolution Mail
Desktop
Azenis.tar.gz
Deleted Items
The Widget Factory
Theme
ComboBoxEntry
ComboBoxEntry
GtkCombo
GtkCombo (disabled)
GtkEntry
GtkEntry
Button1
Button2
ToggleButton1
ToggleButton2
ComboBox
ComboBox
OptionMenu
OptionMenu
CheckButton1
CheckButton2
CheckButton3
CheckButton4
RadioButton1
RadioButton2
RadioButton3
RadioButton4
Move in Harmony
Column1 Column2
Frame (Shadow In)
Frame (Shadow Out)
Frame (Shadow Etched In)
Frame (Shadow Etched Out)
Tab1 Tab2 Tab3
Audacious
1. She Sells Sanctuary (4:13)
128 kB/s 44 kHz
1 She Sells Sanctuary 4:13
2 Fire Woman 5:08
3 Li' Devil 2:46
4 Spiritwalker 3:13
5 The Witch 4:19
6 Revolution 4:17
7 Love Removal Machine 4:19
8 Rain 3:56
9 In the Clouds 4:01
10 Coming Down 4:03
11 Edie (Caio Baby) 4:02
7 Items
1. She Sells Sanctuary ...
The Widget Factory

Why?

- This project started properly 6+ years ago during SUSE’s Hackweek to provide a better out of the box system monitor for the desktop.
- But I have probably been working on it significantly longer

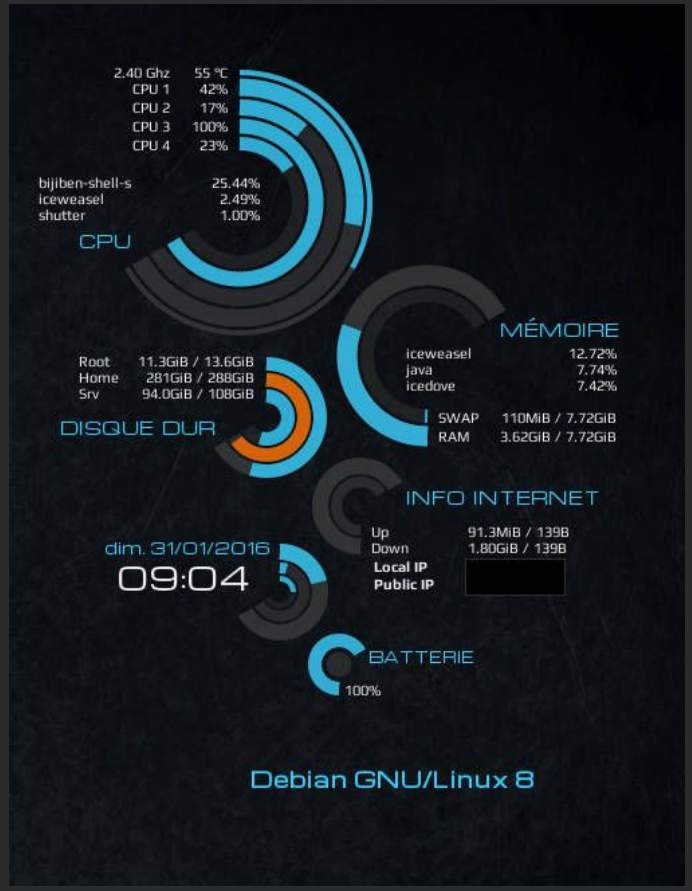
What is conky?

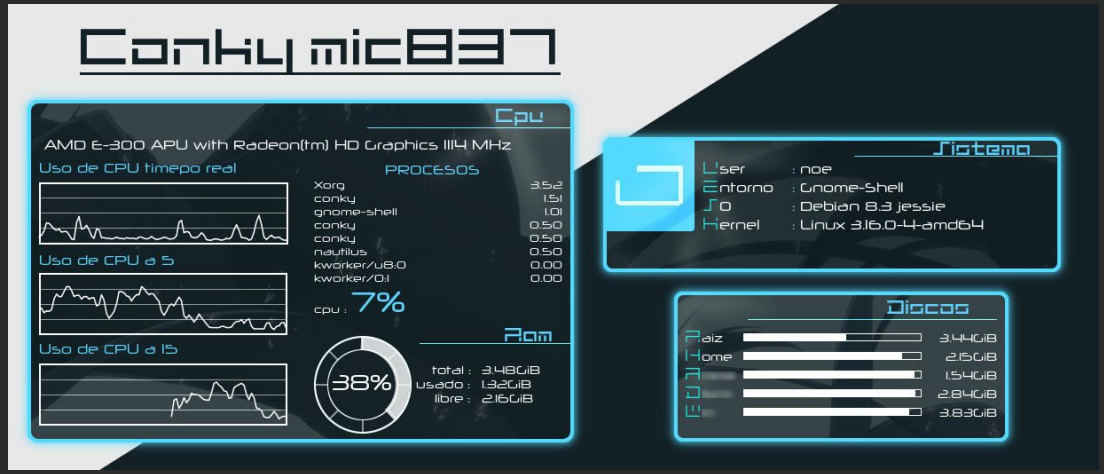
Default


phuket - Linux 2.6.12-gentoo-r8 on x86_64
Uptime: 2h 27m - Load: 2.36 2.56 2.54
CPU Usage: 99%
RAM Usage: 467M/991M - 47%
Swap Usage: 2.7M/1.86G - 0%
Processes: 157 Running: 2
Networking:
Down: 74 k/s
Up: 34 k/s
File systems:
/ 203.90G/465.76G
Temperatures:
CPU: 47.0C - MB: 34.0C
MPD: Pink Floyd - Hey You
Playing
Name PID CPU% MEM%
gaim 1641 2.04 2.31
X 4214 1.02 9.82
mpd 15310 1.02 0.63
gaim 17945 0.00 2.33
Mem usage
java 18234 0.00 14.16
X 4214 1.02 9.82
firefox-bin 5776 0.00 4.74
Current conditions: 7C Scattered clouds

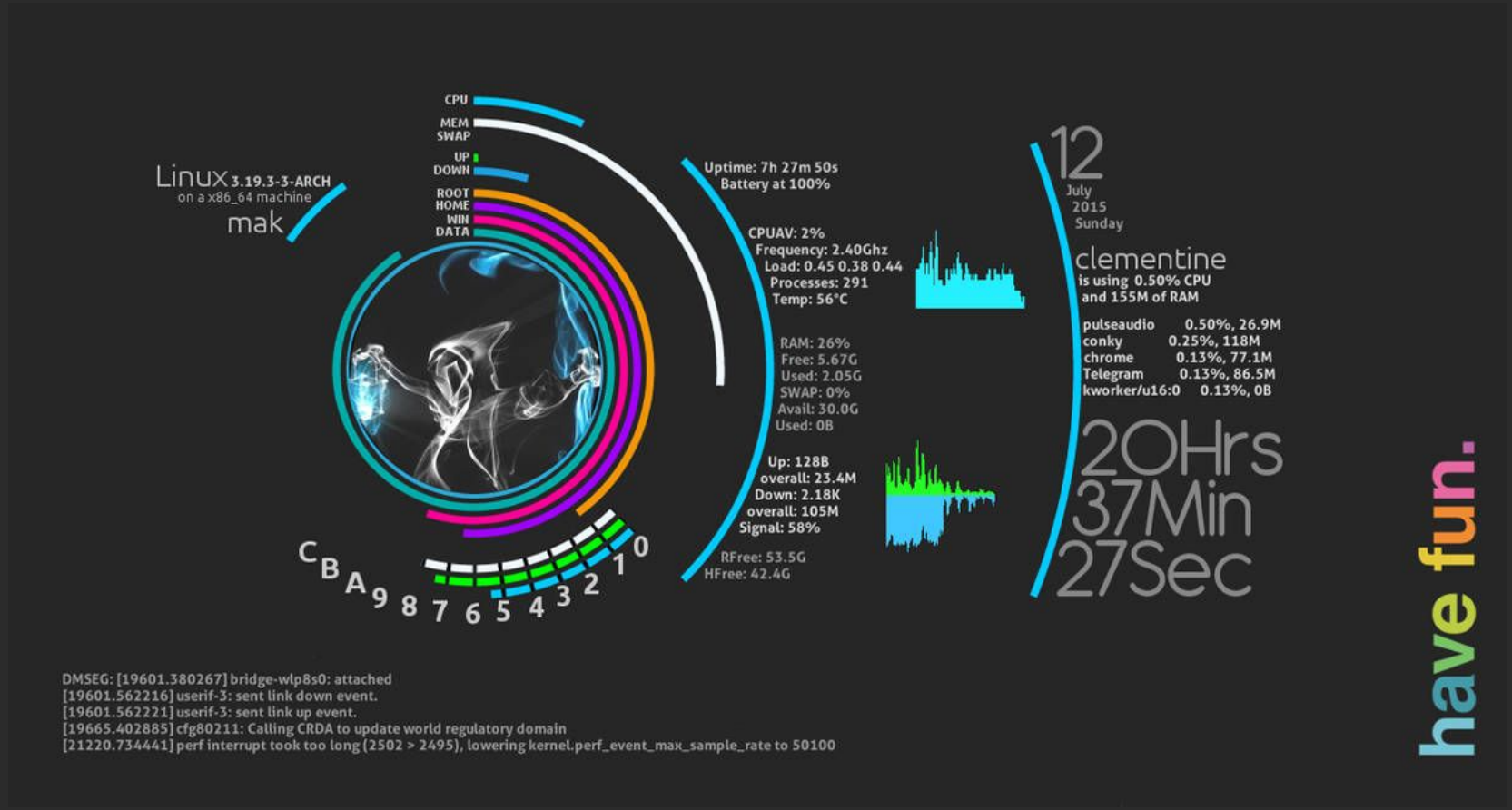
Can look like

Can look like





Can look like



Why do we need something else?

Why?

- Conky Configs are very machine specific
  - Hard to share across multiple machines
  - Even harder for a distribution to provide meaningful defaults

Appearance

- Can look dated out of the box compared to what is possible
- Styling is hard
  - Changing a font requires manually repositioning everything

Why use conky at all?

- The System monitoring works really well, wide range of variables
- Lua integration makes it easy to expand

Goals

Goals

- Work out of the box in Linux Distro’s
- Automated hardware detection
- Theme engine
- Better layout handling

What already exists?

Option 1

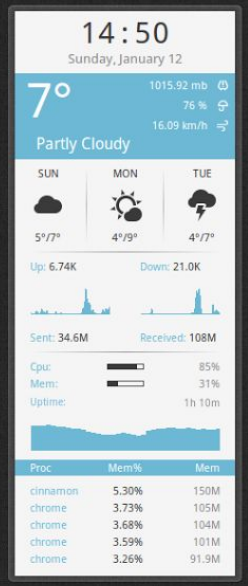

ConkyBar - V3.0
Spectrum Audio by Impulse Screenlet (See description)
18:53:48
16 juli 2015
donderdag
Arch Linux.
Alternatives
(included)
Appearance changes if banshee is running or not.
Banshee -> ON
Banshee -> OFF
Static audio spectrums
(No banshee integration)

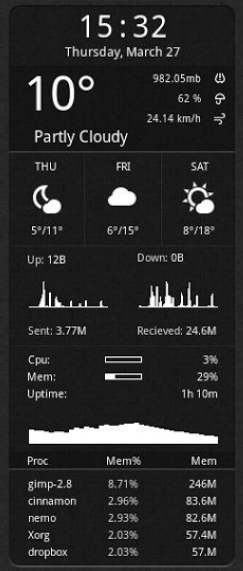
Option 1

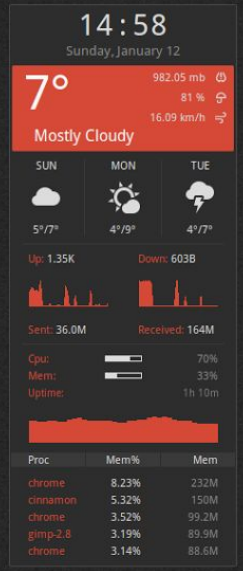
- https://www.deviantart.com/n00by4ever/art/ConkyBar-Conky-config-Conky-1-9-549312244
- Had used it before
- Uploaded once to deviant art
- Just enough scripting and Lua to be code rather then config
- No license
- Icons and fonts weren't licenced for commercial use
- Solved some problems using multiple conky instances

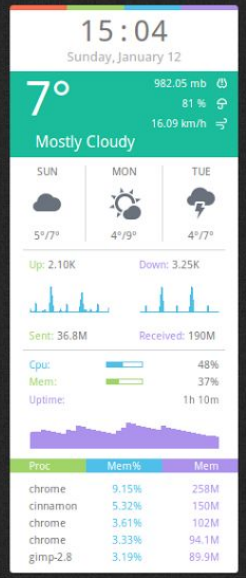
Option 2 - Harmattan

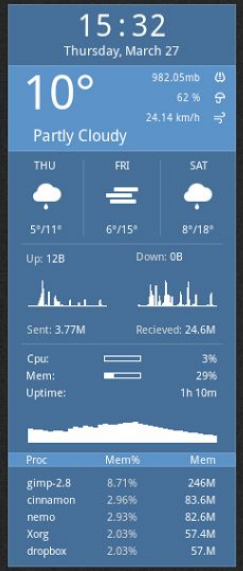
Plethora of themes







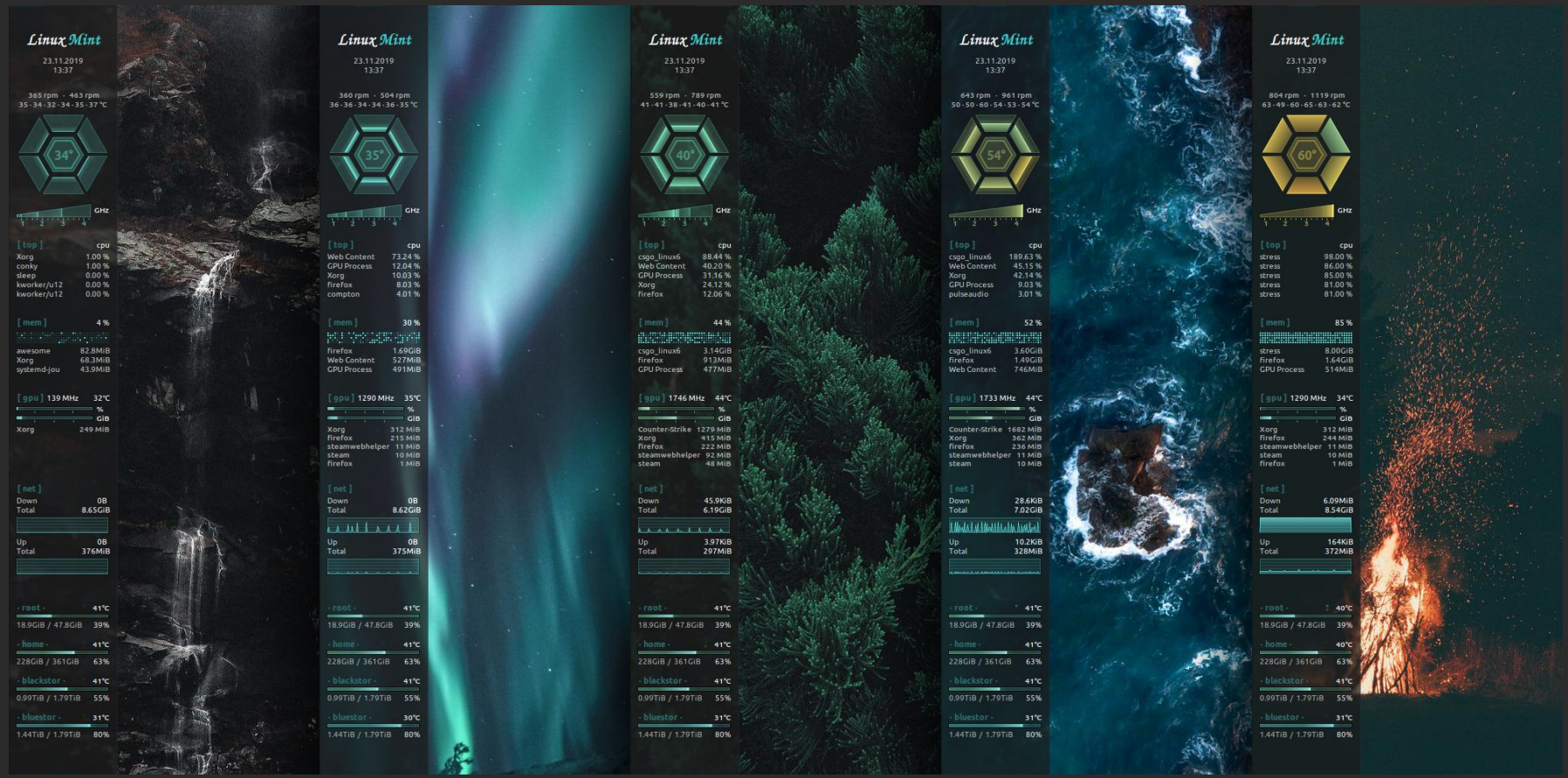




Option 2

- https://github.com/jonasjuffinger/Harmattan
- Looks nice
- Open source licensed
- Doesn't solve the hardware detection problem
- Doesn't solve the layout issues

Option 3 - polycore

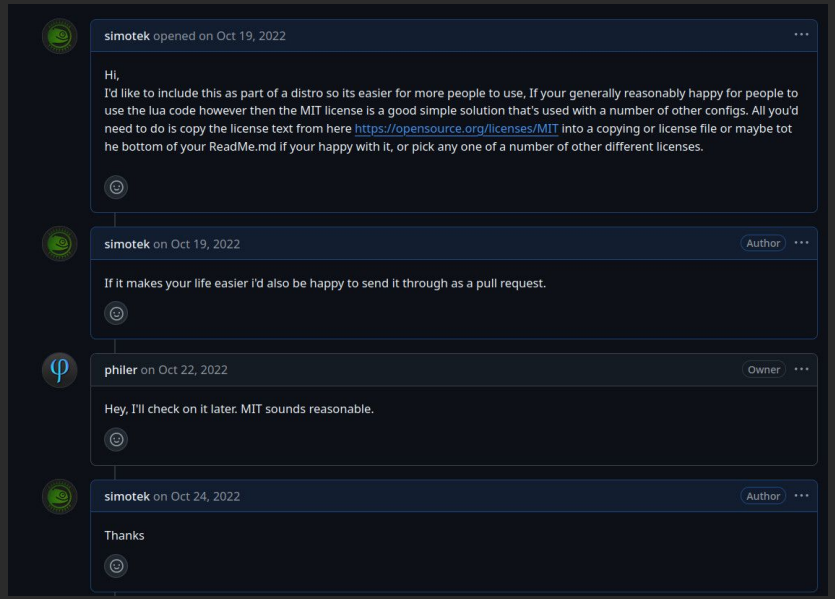


Option 3 - polycore

- Layout engine
- Cool widgets
- Extensive Lua libraries
- No license (yet)
- Still used conky’s traditional text

Option 3 - polycore

- Fixing the License issues
- https://github.com/philer/polycore/issues/5



simotek opened on Oct 19, 2022

Hi,
I'd like to include this as part of a distro so its easier for more people to use, If your generally reasonably happy for people to use the lua code however then the MIT license is a good simple solution that's used with a number of other configs. All you'd need to do is copy the license text from here https://opensource.org/licenses/MIT into a copying or license file or maybe tot he bottom of your ReadMe.md if your happy with it, or pick any one of a number of other different licenses.

simotek on Oct 19, 2022 Author

If it makes your life easier i'd also be happy to send it through as a pull request.

philer on Oct 22, 2022 Owner

Hey, I'll check on it later. MIT sounds reasonable.

simotek on Oct 24, 2022 Author

Thanks

Font Rendering

Freetype / Harfbuzz / Cairo

- https://github.com/anoek/ex-sdl-cairo-freetype-harfbuzz/
- https://github.com/brndnmtthws/conky/pull/1501

When is it time to fork a project?

Why fork

- Aims and goals no longer align
- elopment cycle differencesDifferences in developer time

Introducing conky-bubbles

conky-bubbles

- Conky was named after a character in the Trailer Park Boys
- A wiki suggested that Bubbles, was conky’s helper.

conky-bubbles

- Features
  - Improved layout engine with sandboxing
  - Widgets render text using the new text API
  - Theme Engine
    - Dimensions
    - pcore2

Still to do

- More Themes
- Further Improve hardware detection
- More Widgets
- Launcher

Compiling conky

mkdir build; cd build
cmake .. -G Ninja -DCMAKE_INSTALL_PREFIX=/opt/conky -DBUILD_APCUPSD=ON
-DBUILD_ARGB=ON -DBUILD_CURL=ON -DBUILD_DOCS=OFF
-DBUILD_EVE=ON-DBUILD_GUI=ON -DBUILD_HDDTEMP=ON -DBUILD_HTTP=OFF
-DBUILD_I18N=ON -DBUILD_IBM=ON -DBUILD_ICAL=OFF -DBUILD_ICONV=ON
-DBUILD_LUA_CAIRO=ON -DBUILD_IMLIB2=ON -DBUILD_LUA_IMLIB2=ON
-DBUILD_LUA_RSVG=ON -DBUILD_LUA_TEXT=ON -DBUILD_IOSTATS=ON
-DBUILD_IPV6=ON -DBUILD_MATH=ON -DBUILD_MOC=ON -DBUILD_MYSQL=OFF
-DBUILD_NCURSES=ON -DBUILD_NVIDIA=ON -DBUILD_PORT_MONITORS=ON
-DBUILD_PULSEAUDIO=ON -DBUILD_RSS=ON -DBUILD_OLD_CONFIG=ON
-DBUILD_WEATHER_METAR=ON -DBUILD_WEATHER_XOAP=ON
-DBUILD_WLAN=OFF -DBUILD_X11=ON -DBUILD_XDAMAGE=ON -DBUILD_XDBE=ON
-DBUILD_XFT=ON -DBUILD_XSHAPE=ON -DOWN_WINDOW=ON
-DBUILD_XMMS2=OFF
ninja; ninja install

Questions

https://github.com/simotek/conky-bubbles

Spec Files - Example

- https://build.opensuse.org/package/show/devel:languages:python/python-bytecode
- https://build.opensuse.org/package/show/devel:languages:python/python-autobahn

Step 1: Understand your Package

Step 2: Try and find a similar Package

Autogenerate

- For some languages such as python there are tools to autogenerate a spec file
- You can also create a empty spec file with tools like “vim foo.spec”

Branch or Create

- If we are working on a package that already exists we can branch it into our home project
- Otherwise we need to create it.

Finding The right version

- Browse to openSUSE:Factory
  - https://build.opensuse.org/project/show/openSUSE:Factory
- Use the filter to find the package you want
- Select the package
- At the top right look for Developed at XXX
- This will give you the devel project.
- Click on it
- Select Branch Package on the Left.

Creating a Package

- Browse to your home project (not mine)
  - Shortcut on Left Bar
  - https://build.opensuse.org/project/show/home:simotek
- Using osc
  - osc co project
  - cd project
  - osc mkpac name
  - cd name

Step 3: Building

Working Locally

- First we need to checkout the package
  - osc co project/package
- Then we can work with the files locally
- To add / remove files from the project
  - osc ar (Similar to git add)
- Commit changes when done
  - osc commit Similar to git commit + git push

Building

- First configure repositories.
  - https://build.opensuse.org/repositories/home:simotek
- osc build repo
- The buildservice will automatically build any enabled packages.

Step 4: Contributing

Contributing

- Send from “home” project to “devel” project
- “devel” project maintainer will send it to Tumbleweed
- You can then send it to the next Leap
- This can be done either with a submit request from the webui or on the command line with osc sr

Staging Dashboard

- https://build.opensuse.org/staging_workflows/openSUSE:Factory

Maintenance

- Sometimes we need to change a package in an existing Leap distribution
- osc mbranch and osc sr when done

2 Minutes of Advanced Concepts

Constraints

- Specifies minimum requirements for a builder
- _constraints file
- https://openbuildservice.org/help/manuals/obs-user-guide/cha.obs.build_job_constraints.html
- Example
  - https://build.opensuse.org/package/view_file/LibreOffice:Factory/libreoffice/_constraints?expand=1

rpmlintrc

- Rpmlint does a good job of finding issues in packages sometimes it creates false positives though
- <packagename>-rpmlintrc file
- https://en.opensuse.org/openSUSE:Packaging_checks
- Example: https://build.opensuse.org/package/show/Base:System/python-dbus-python

Supplements

- You have a vim plugin for Rails, or a
- Squid plugin for yast
  - Supplements: (A and A-Client)
  - Supplements: (vim and ruby-common-rails)

Multibuild

- Build multiple packages from the same spec and source
- _multibuild file
- https://openbuildservice.org/help/manuals/obs-user-guide/cha.obs.multibuild.html
- Example: https://build.opensuse.org/package/show/devel:tools:building/cmake

Project Config

- Used to define settings between different Distributions along with some rpm macro’s

Language files

- Header
  - %lang_package
- Install
  - %find_lang %{name}
- Files
  - %{?suse_version:%files lang -f %{name}.lang}
- Example
  - https://build.opensuse.org/package/view_file/X11:Enlightenment:Factory/terminology/terminology.spec?expand=1

Packaging Workshop

Simon Lees
Simotek

sflees@suse.de / simon@simotek. net
Simotek on irc.libera.net / Discord
@Simotek_Dot_Net

A Hackweek Experement

https://en.opensuse.org/openSUSE:ALP/Workgroups/GrassyKnoll

Getting Started

- Create an account at https://idp-portal.suse.com/univention/self-service/#page=createaccount if you don’t have a bugzilla / obs / openSUSE Community account.
- Install patterns-devel-osc-devel_osc_build (This should work with a WSL Image as well)
- osc co home:<your-user>

What are we doing?

- Has anyone done packaging before?
- Has anyone got any questions or something they'd like to look at?
- We can also look at building Images if you'd like.

Breaking a Package

- Add a file that doesn’t exist
- Remove one that does
- Make a typo in a macro name