

Meet the Cat: Pd-L2Ork and its new cross-platform version Purr Data

Ivica Ico Bukvic <ico@vt.edu>

Albert Gräf <aggraef@gmail.com>

Jonathan Wilkes <jon.w.wilkes@gmail.com>

May 2017



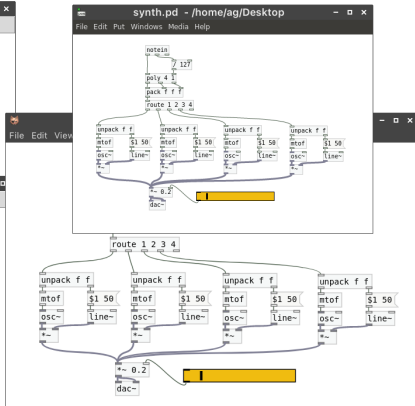
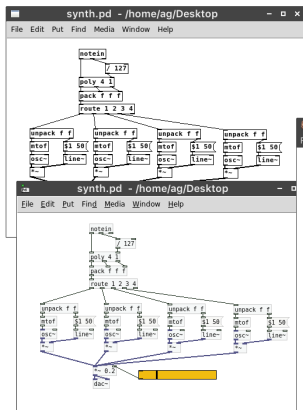
JOHANNES GUTENBERG
UNIVERSITÄT MAINZ

Miller Puckette's Pd a.k.a. Pure Data

- ▶ *open-source* (3-clause BSD) alternative to Max
- ▶ graphical *data flow programming* environment
- ▶ geared towards real-time interactive *computer music* and *multimedia* applications
- ▶ rock-solid *deterministic* real-time audio engine
- ▶ runs on *Linux, Mac* and *Windows*
- ▶ community *flavors* and *distributions* (Pd-extended, Pd-l2ork)



The Different Flavors



Pd: A Brief History

- ▶ **Pd** (Pure Data, “vanilla”, Miller Puckette et al): 1996 – ongoing
- ▶ **Pd-extended** (Hans-Christoph Steiner): 2002 – 2013
- ▶ **Pd-l2ork** (Ico Bukvic): 2010 – now in maintenance mode
- ▶ **Purr Data** (Jonathan Wilkes): 2015 – ongoing

Purr Data = Pd-l2ork 2.x = Pd-extended’s “spiritual successor”
first alpha release April 2016, announced on the [Pd forum](#)



Pd-l2ork at JGU

- ▶ learned about it through **Dave Phillips' 2012 Linux Journal** article
- ▶ been used heavily at JGU since about 2013 as **vanilla replacement**, mainly for the improved GUI and editing features
- ▶ 2015: Jonathan starts working on a **Mac/Windows port**, rumors on the DISIS mailing list
- ▶ winter semester 2016: **pilot run** of Purr Data at JGU, teach in the day, hack at night
- ▶ resulted in **2.0** and **2.1** versions, not bug-free *yet*, but getting there, and perfectly usable for teaching purposes already



Installation

- ▶ **Arch:** AUR or JGU repo, see <https://l2orkaur.bitbucket.io/>
- ▶ **Ubuntu:** JGU PPAs on Launchpad, see <https://l2orkubuntu.bitbucket.io/>
- ▶ **Mac, Windows, Linux (Debian):** <https://github.com/agraef/purr-data/releases>
- ▶ **Github mirror:** <https://agraef.github.io/purr-data/>
- ▶ **Jonathan's Gitlab:** <https://git.purrrdata.net/jwilkes/purr-data>
- ▶ **Caveat:** building from source takes approx. 20 min, *lots* of dependencies required!



Goals

- ▶ better GUI and editing features (Pd's GUI works, but isn't convenient)
- ▶ “batteries included”, large bundled collection of popular externals: Gem, cyclone, iemlib, lyonpotpourri, mrpeach, ...
- ▶ special features tailored to the Linux Laptop Orchestra (L2Ork)
- ▶ “nimble” development model, quickly improve things and fix bugs even if that might break backward compatibility in some places (-legacy option helps if 100% backward compatibility is needed)



Why Purr Data?

- ▶ **cross-platform** (Pd already is, but not Pd-l2ork)
- ▶ Pd-l2ork needs some Tcl extensions (TkPath, Tcl Xapian) not readily available on modern Mac, Windows
- ▶ Tcl/Tk is getting **old**, programming in Tcl is easy but has its limitations
- ▶ switch to a better, more **modern GUI** as well as a **better language and toolkit** was in order



The Solution: nw.js

- ▶ **nw.js** = node-webkit = JavaScript runtime (*NodeJS*) + browser engine (*Chromium*)
- ▶ possible alternatives: **Electron**, native cross-platform GUI toolkit
- ▶ GUI programmed in **JavaScript**, opening the possibility to utilize modern web technology (HTML5, CSS3, SVG)
- ▶ **as before**: socket-based communication (“FUDI”) between GUI and real-time engine



Pd-l2ork Goodies

- ▶ graphical improvements (SVG graphics)
- ▶ improved editing features (graphical resizing of GUI elements and GOP areas, aligning objects, ...)
- ▶ infinite undo
- ▶ intelligent patching
- ▶ “K12 mode” for pre-college school education (not in Purr Data yet)



Purr Data Goodies

- ▶ better zooming
- ▶ working print option
- ▶ new help browser
- ▶ improved SVG graphics and data structures
- ▶ easier development (JavaScript) and debugging (integrated developer console)



Demo

- ▶ improved editing
- ▶ infinite undo
- ▶ intelligent patching
- ▶ zooming
- ▶ theming
- ▶ printing
- ▶ help browser
- ▶ SVG graphics and data structures
- ▶ Pure+Faust (if there's time)



So Where Do We Stand?

- ▶ **GUI port** basically finished now, but ...
- ▶ ... some of the **external GUIs** still need to be ported
- ▶ follow **upstream** for the latest changes in the engine
- ▶ **bugfixes**, e.g., out-of-order FUDI messages, inefficiencies in some of the SVG graphics
- ▶ tackle **concurrency** (pd~, native support for managing multiple real-time engines)
- ▶ **SVG graphics and data structures**, make them GOP-compatible \Rightarrow facilitate the creation of custom GUI elements
- ▶ need people who use Purr Data for their daily work and then **report bugs, add documentation, contribute code** etc.

