

Workshop: Getting Started with Purr Data

Albert GRÄF

Dept. of Music-Informatics, Johannes Gutenberg University (JGU)
Mainz (Germany),
aggrae@gmail.com

Abstract

This is a hands-on workshop for all Linux audio users interested in real-time interactive signal processing in general and Jonathan Wilkes' new "Purr Data" flavor of Miller Puckette's Pd in particular.

Keywords

data-flow, pd, pd-l2ork, pd-lua, purr-data

Introduction

Jonathan Wilkes' Purr Data a.k.a. Pd-l2ork 2 was first presented at PdCon 2016 [2] and LAC 2017 [1]. It is an improved version of Miller Puckette's Pd, based on Ico Bukvic's Pd-l2ork. Sources and binary packages for Arch, Debian and Ubuntu are available at Purr Data's official Github mirror.¹

Pd is an open-source graphical data-flow programming environment which is geared towards real-time interactive computer music and multimedia applications. Purr Data serves the same purpose, but offers a new and much improved graphical user interface and includes many 3rd party plug-ins. Like Pd, it runs on Linux, Mac and Windows, and is open-source throughout.

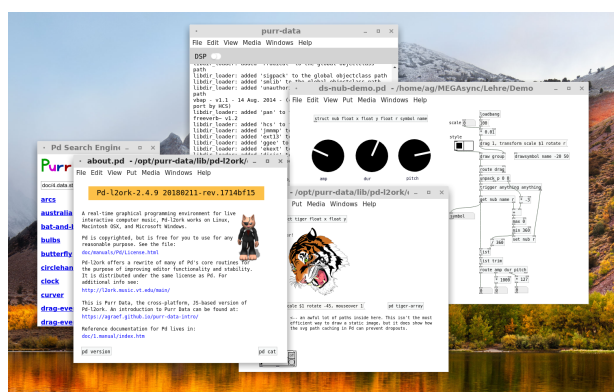


Figure 1: Purr Data running on Linux.

About the Workshop

This 90 min workshop is intended for Linux audio users who want to get started with Purr Data. For an introduction to Purr Data, please check our LAC17 presentation² and the Quick Introduction.³ Topics of the workshop will proceed from the basics to the moderately advanced and may include (as time permits):

- Getting a minimal Purr Data installation up and running quickly on Linux (compiling from source).
- First-time setup (configuring MIDI and audio devices, setting up Pd-Lua).
- Real-time audio and MIDI processing, making music with Purr Data.
- Programming new objects (a.k.a. Pd externals) the easy way with Pd-Lua.⁴
- Custom graphics and data structure visualizations in Purr Data.

Prerequisites

If you want to follow along with the installation, please make sure to have the following development tools installed: gcc, GNU make and git. You'll also need ALSA and Jack (including the corresponding development packages if needed), and Lua 5.3 (for external programming). A Jack/MIDI patchbay like Qjackctl and software MIDI synthesizer like Qsynth will also come in handy. These should all be readily available in most Linux distributions.

For the programming section it helps to have at least a passing familiarity with the Lua scripting language.⁵

²<https://www.youtube.com/watch?v=T1wo496Zx0s>

³<https://aggrae.github.io/purr-data-intro>

⁴<https://github.com/aggrae/pd-lua>

⁵<https://www.lua.org>

¹<https://aggrae.github.io/purr-data/>

References

- [1] I. Bukvic, A. Gräf, and J. Wilkes. Meet the cat: Pd-L2Ork and its new cross-platform version “Purr Data”. In *Proceedings of the 15th International Linux Audio Conference*, pages 141–148, St. Etienne, 2017. Jean Monnet University.
- [2] I. Bukvic, J. Wilkes, and A. Gräf. Latest developments with Pd-L2Ork and its development branch Purr-Data. PdCon 2016, New York, NY, USA. http://ico.bukvic.net/PDF/PdCon16_paper_84.pdf, 2016.