Live coding with Makkeróni - workshop

Balázs Kovács, dr. habil.

University of Pécs, Faculty of Arts, Department of Electronic Music and Media

H-7630 Pécs, Zsolnay Vilmos u. 16.

kovacs.balazs@pte.hu

ABSTRACT

Makkeróni is a web-based live coding system which simulates a linux shell. The user operates with commands known from the BASH (ls, help, cat etc.), and also there are some audio-specific commands (play, freq, tone etc.). In this workshop we're going to learn how to use it, while possibly focusing on efficiency (pipes, joker characters, command history tricks) and network cooperation. If the participants are open for it, we could discuss the targets of further development as well.

It's recommended for both new and experienced Linux shell users.

Optimal length: 45-70min

Link to the application: http://makker.hu/makkeroni/

1. INTRODUCTION

The idea of Makkeróni came from two directions: on one side, I've been working on linux system modifications for creating the running processes audible (it's called Proc filesystem music[1]), inspired by the SonicFinder project by William W. Gaver (1989). In this case my insterest contiously focused on low-level approaches for sound synthesis or processing, trying to use system-level processes in-stead of auditory icons. On the other side I began to create inter-website or real-world ↔ web communication projects, for example a webpage-controlled mechanical device or public light or-gan projects on the chimneys of the Zsolnay Cultural Quartier, Pécs. These projects opened op the way for starting cross-website multimedia interaction.

Collecting something from all of the above projects, the idea of a web-based sonic operating system was born: a web application that simulates a linux shell - the operating system which is run-ning the webserver itself - while the user is controlling it with commands and their arguments on the client's side. The result is Makkeróni, a text-based audio "operating" system.

[1] Proc filesystem music, as documented at the following URL: http://kbalazs.periszkopradio.hu/index.php?file=works/2005-lad

While designing Makkeróni, the following principles were used:

1: create a simple-to-use, easy-to-run live coding system with zero or minimal external libraries; create an useful and practical interface, which makes easy to seamlessly learn the use of the Linux shell;

2: create an audio system which utilizes the efficiency of linux: not only with the command and file structure, but with the low-level approach to processes, data etc. as well. It could be a goal also to create a system, which could be a virtual terminal too for a real, locative server;

3: utilize the native possibility of the web: two-way communication between client and server or client and other clients, data sharing etc.;

I tried to achieve these principles with the following methods:

- Makkeróni is much simplier to use than other live coding systems. It cames from the shell behaviour: one line + enter key = one command. Everything is self-documented in a practical way. The user can be quickly introduced into the application while simply using and interacting with it: help and other command works in order to explain the use. Changelog, reference and other documents are also in-system.

- using existing shell commands (ls, cat, ps etc,) include the operating system commands into the live coding process - which means also that they produce sounds also by running, creating sounds which are reflecting to the result of the command itself;

- using joker characters to randomize parameters;

- using command history to quickly recall the previous events;

- using autocomplete for efficiently typing of the commands;

- the system should be capable to be expanded with user-added sounds and presets, available for the other users of Makkeróni;

- the users can save the running processes' list on the server, letting them possible for others to use their results.

Makkeróni has been realized with adapting the JQuery Terminal Emulator by Jakub Jankiewicz,[2] which offers many possibilities to create a shell-like interpreter, parsing commands and arguments, provides mobile-friendly interface etc. As a

[2] https://terminal.jcubic.pl/

starting point, I expanded this wonderful framework with extra functions that could perform the audio processing and live controlling. In the workshop, I'll going to present how it works.
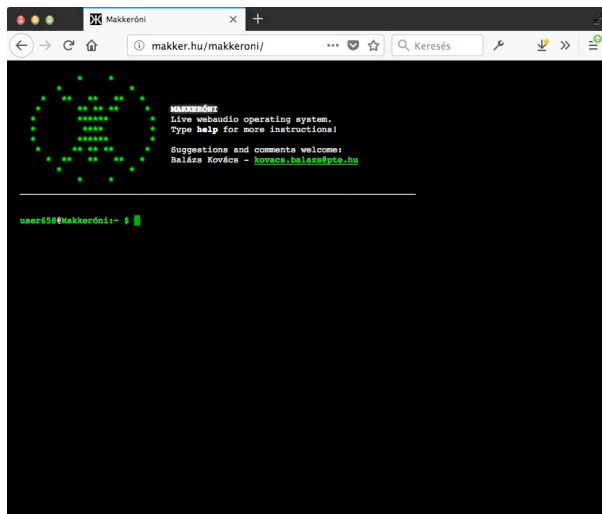

Figure 1. Makkeróni start-up screen. Type 'help' for getting started!

2. SHORT REFERENCE

In the followings I summarize the main commands of Makkeróni in the current phase of development:

Table 1. Short reference

Command	Description
freq	play a sinewave tone
fmfreq	play an fm-modulated sinewave tone
tone	play a simple tone with filter
makkeróni	fm-modulated synth with audio rate mathematical pair
ls	list contents of the home, soundfiles & saved presets folders
play	play a soundfile
loopplay	play a soundfile looped
fadeplay	play a soundfile looped, with linear fade-out
watch	repeatedly start play, freq, fmfreq or any other commands
batch	start a command in multiple instances
seq	store a number of sequences into one of the 4 sequence slots
seqlist	simple stepsequencer for lists of numbers stored by the "seq" command
stop	stop one or more loop-play thread or watch process

sleep	sleep (mute) one, more or all processes
resume	resume (unmute) a process
remakker	restart a loop-play or watch thread. Provides new process id
replace	replace a thread with a new command
ps	list of running loopplay and watch processes
degrade, bitshift	sample-level audio manipulation (LOUD!)
upload	upload a sample (wav,mp3 or ogg) into the soundfile folder
help	short description of commands
fontsize	set fontsize (default: 12)
statuslength	set number of lines in status bar (default: 7)
clear	clear window
cat	print the contents of a textfile
connect, disconnect	connects or disconnects to/from the chat server
wall	send a command or message to the other users

In addition of the above, there are some general syntax to be applied for most of the commands:

- *: random number (randomize parameters on play, loopplay, fadeplay, freq, and fmfreq)

- ↑ and ↓ arrows: browse command history

- something and TAB key: autocomplete command (for. ex 'lo' + TAB gives 'loopplay' back)

3. ADDITIONAL INFORMATIONS

The workshop is open for anybody interested in using linux shell for creating audio or multimedia processes. The active participants need a computer, I'll need a projector and PA. And of course we all need network access:) If You're interested, please visit Makkeróni's website: http://makker.hu/makkeroni/. If You have recommendations for a special topic to cover, please drop me a line in advance, to the email address above. And, if You have ideas to implement, and/or time to help in further development, You're mostly welcome!

4. BIO

Balázs Kovács, PhD habil., philosopher, holds a PhD in aesthetics of interactive sonification. Head of the Electronic Music and Media Arts programme at University of Pécs, Faculty of Arts. Founder of the Hangfarm (Soundfarm) open-air media art exhibition place in Ellend, Hungary.

5. REFERENCE

[1] Kovács, B. 2019. Introducing Makkeróni. *International Conference on Live Coding.* *http://iclc.livecodenetwork.org/2019/papers/paper62.pdf*