



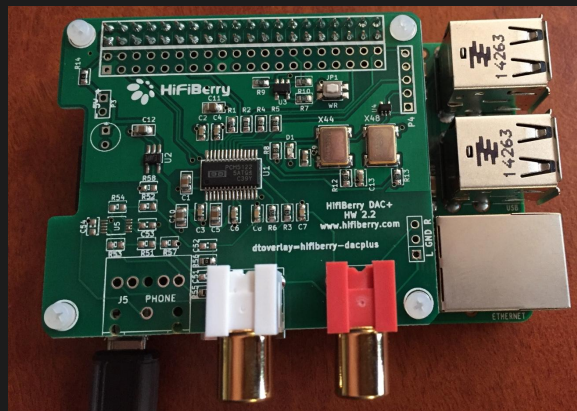
ADC²⁴
Bristol

**SQUEEZE JUICE AND BOTTLE IT
INTO EMBEDDED DEVICES
AND MORE**

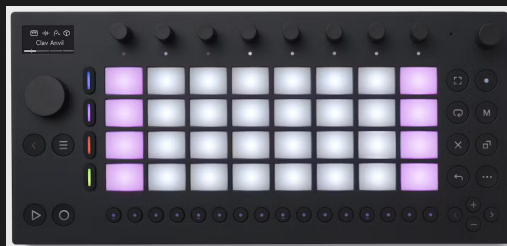
STEFANO ZAMBON

Embedded Linux Audio

Hobby Projects...



... and professional products



Hobby vs Industrial

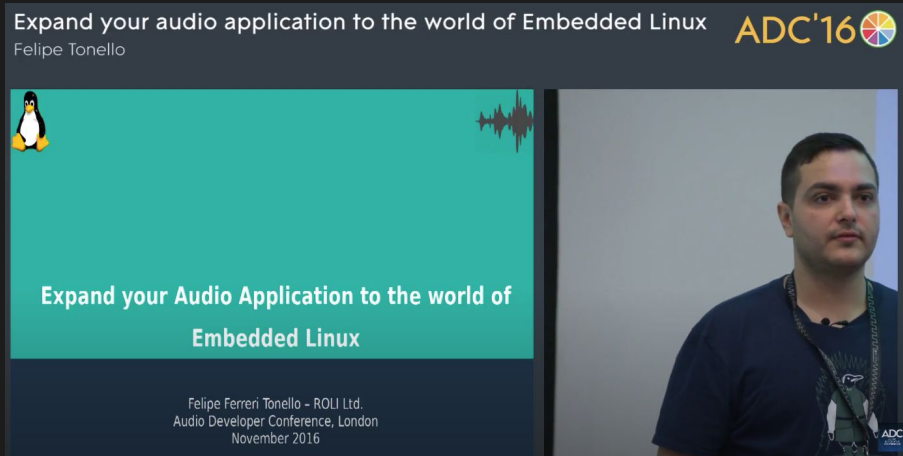
" Why can't I just use e.g. Raspberry Pi OS? "

- Power-off safety
- Easy & robust update system for end users
- Boot time and size optimization
- Reproducible builds
- Support other HW platforms

Past talks (ADC16 & ADC19)

Expand your audio application to the world of Embedded Linux **ADC'16**

Felipe Tonello

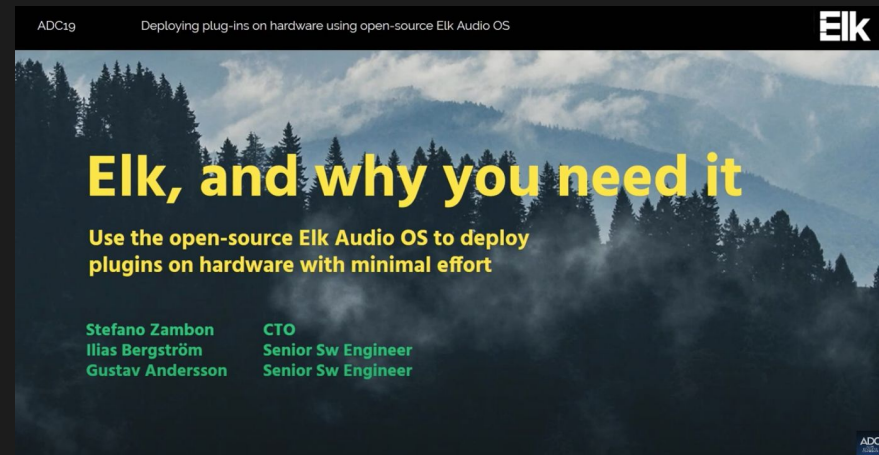


Expand your Audio Application to the world of Embedded Linux

Felipe Ferreni Tonello - ROLI Ltd.
Audio Developer Conference, London
November 2016

Elk Audio OS

ADC19 Deploying plug-ins on hardware using open-source Elk Audio OS **EIK**



Elk, and why you need it

Use the open-source Elk Audio OS to deploy plugins on hardware with minimal effort

Stefano Zambon
Ilias Bergström
Gustav Andersson

CTO
Senior Sw Engineer
Senior Sw Engineer

In the meantime...

Major changes in JUCE (esp. CMake support)

Suggested solutions not supported anymore

New versions of Yocto & new tools in the ecosystem

Yocto / OpenEmbedded

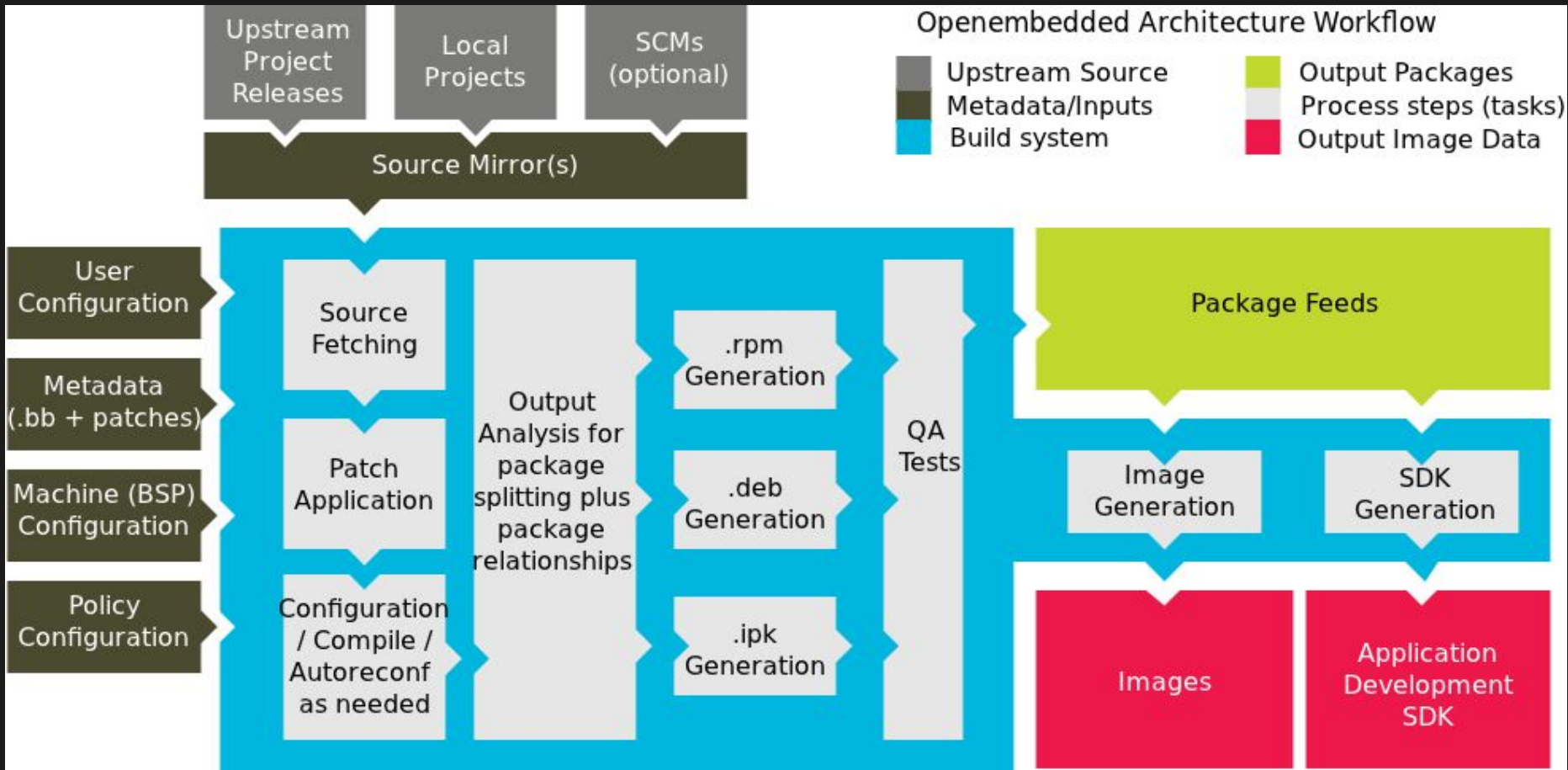


<https://www.yoctoproject.org/>

*“It’s not an embedded Linux distribution,
it creates a custom one for you.”*

Industry-standard tool to create a Linux BSP (Board Support Package)

(simpler but less powerful alternative: Buildroot)



What you get / 1

"Target" image:

- Bootloader
- Kernel & Device Tree
- Root filesystem
 - Your selection of Linux standard components
 - Libraries & products specific for your product

What you get / 2

- Cross-compiling SDK
- Multiple configurable images (development, production, etc)
- QEMU targets for automated testing
- License manifesto for all the components

How you get it / 1

By writing *Bitbake recipes*:

```
1 SUMMARY = "MDA VST3 plugins for Linux"
2 HOMEPAGE = "http://mda.smartelectronicx.com/"
3
4 LICENSE = "GPLv3"
5 LIC_FILES_CHKSUM = "file://COPYING;md5=e49f4652534af377a713df3d9dec60cb"
6
7 PV = "0.1.0+${SRCREV}"
8 SRC_URI = "\
9     git+https://github.com/elk-audio/mda-vst3;protocol=https;nobranch=1 \
10    file://0001-Added-DNDEBUG-compile-definition.patch \
11    file://0001-Fix-for-gcc-10.patch \
12    file://0002-Removed-auto-strip-with-Release-build-as-Yocto-takes.patch \
13 "
14
15 SRCREV = "5b970765c49480880e6c945de7baf040ff703c50"
16
17 S = "${WORKDIR}/git"
18
19 inherit cmake
20
21 OECMAKE_C_FLAGS_RELEASE += "-O3"
22 OECMAKE_CXX_FLAGS_RELEASE += "-O3"
23 EXTRA_OECMAKE = "-DCMAKE_BUILD_TYPE=Release"
24
25 MDA_PLUGIN_DIR = "/home/mind/plugins/mda-vst3"
26
27 do_install() {
28     install -d ${D}${MDA_PLUGIN_DIR}/mda.vst3/Contents/${TARGET_ARCH}-linux
29     cp "${WORKDIR}/build/VST3/Release/mda.vst3/Contents/x86_64-linux/mda.so" "${D}${MDA_PLUGIN_DIR}/
30 }
31
32 FILES_${PN} += "${MDA_PLUGIN_DIR}"
33 FILES_${PN} += "${MDA_PLUGIN_DIR}/*"
```

Think like Makefiles,
at one extra layer of abstraction:

- Where to fetch sources
- Apply patches
- Specific build flags
- Installation for your root fs

Not only for applications, also:
scripts, system services, user config...

How to get it / 2

By creating *OpenEmbedded Layers*:

- Collection of recipes and higher-level config files
- Many already available and maintained:
 - [Open Embedded Layers Index](#)

And creating *target images* by choosing packages from layers

Layers for audio projects

[meta-multimedia](#)

part of the default openembedded-core,
for generic audio playback applications

[meta-musicians](#)

collections of music-production recipes (DAWs,
plugins, etc.) typically used in Desktop Audio Linux

[meta-elk](#)

HW-agnostic parts of Elk, with focus on headless devices
running audio plugins on RT Xenomai kernels

Putting layers together

Managing multiple nested Git repos for each layer
+ project configuration files

KAS is a recent tool by Siemens that makes the job easier

Example: Elk Audio OS image for Raspberry Pi 4

<https://github.com/elk-audio/elk-audio-kas-configs>

Example: Elk for RPi4

HW-independent layers
(shared between multiple architectures)

```
1 # KAS project file
2 header:
3   version: 14
4   includes:
5     - common/elk-audio-os-v1.0.0.yml
6
7 machine: raspberrypi4-64
8 target: elkpi-audio-os-image
9
10 repos:
11   meta-elkpi:
12     url: https://github.com/elk-audio/meta-elkpi.git
13     commit: a262b989b6325250c41073fa7113df1b7364a2d5
14     path: layers/meta-elkpi
15   meta-raspberrypi-elk:
16     url: https://github.com/elk-audio/meta-raspberrypi-elk.git
17     commit: 10ca6cae920cd879fe173ca9f05156f1c9f83521
18     path: layers/meta-raspberrypi-elk
19   meta-raspberrypi:
20     url: https://github.com/agherzan/meta-raspberrypi.git
21     commit: 2a06e4e84b04fc900f3a4524581548c9b5e57362
22     path: layers/meta-raspberrypi
23
24 local_conf_header:
25   elkpi-conf: |
26     SWU_VERSION = "1.0.0"
27     RPI_KERNEL_DEVICETREE:remove = "broadcom/bcm2711-rpi-400.dtb broadcom/bcm2711-rpi-cm4.dtb"
28     RPI_KERNEL_DEVICETREE_OVERLAYS:remove = "overlays/vc4-kms-dsi-7inch.dtbo"
29     ENABLE_I2C = "1"
```

“Product” layer:
top-level application

“Machine” layers:
HW-specific support (kernel etc)

Adding your JUCE plugin

Used to be a little difficult,
especially cross-compilation setup and headless support

Much easier now with recent JUCE & CMake support!

Yocto support for JUCE added in [latest Elk Audio OS SDK](#)
(same recipe could be used without Elk)

... very simple, just add in your recipe `inherit juce`
And that's it!

JUCE plugins caveats

- Only for CMake-based projects (no Projucer)
- Only VST3 targets (no standalone)
- Only headless plugins (GUI needs to be external process)
- Some system resources might be in different locations

Advanced Topics: swupdate

Over-the-Air and USB updates

Robust to power-off at any time

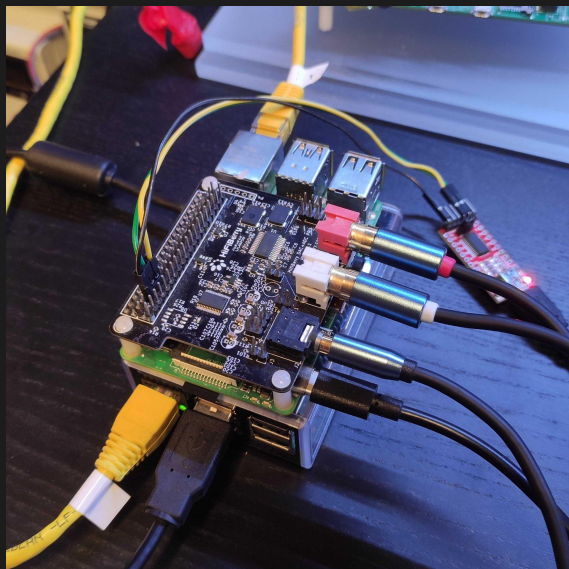
Use a redundant partition scheme

[meta-swupdate](#)

Advanced Topics: tbot

<https://tbot.tools/>

Automated CI tool to deploy and run tests of various sorts on target HW devices connected to a build server



Even bootloader & real-audio tests!

The End!

Questions?



stefano@elk.audio