

Piece of cake – testing remote embedded devices made easy

using open-hardware MuxPi

Samsung R&D Institute Poland, Paweł Wieczorek

Samsung Open Source Conference 2018

Agenda

1. Introduction
2. Previous efforts
3. Idea
4. Hardware
5. Software
6. Next steps
7. Conclusion

Introduction



Use cases



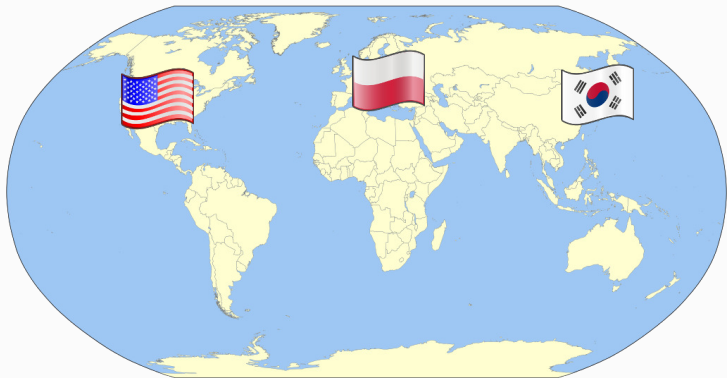
<https://news.samsung.com/global/tizen-4-0-first-milestone-release-to-open-new-opportunities-in-the-iot-era>

- Continuous platform development
- QA step prior pulling new changes
- Package internal tests are **not** enough

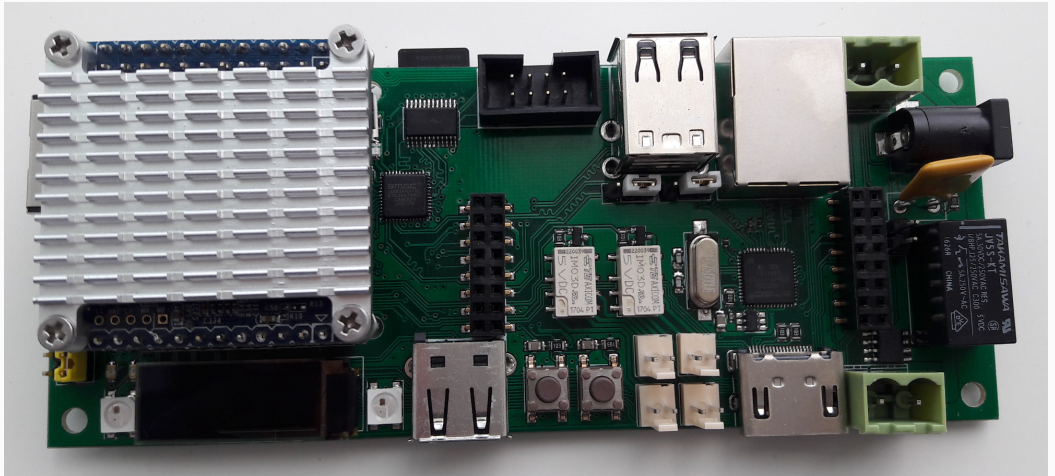


Remote accessibility

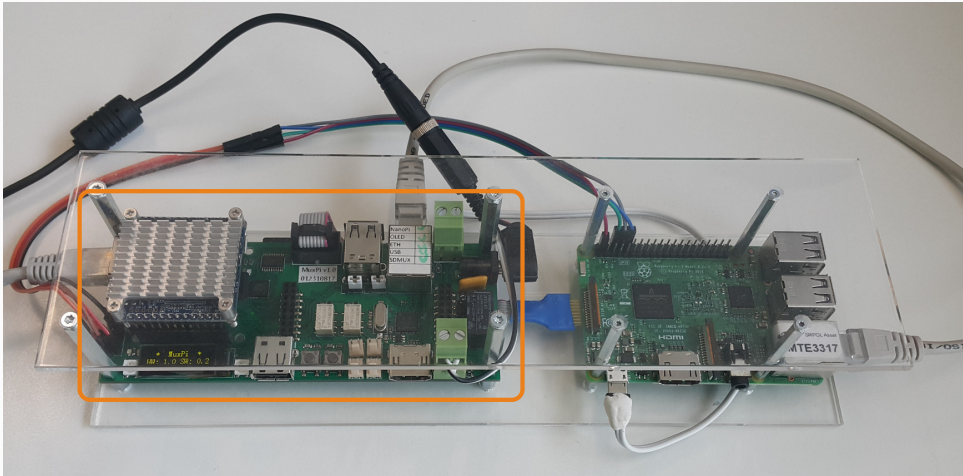
- Easy to store in a secure manner
- Less effort than per developer
- Better utilized when shared



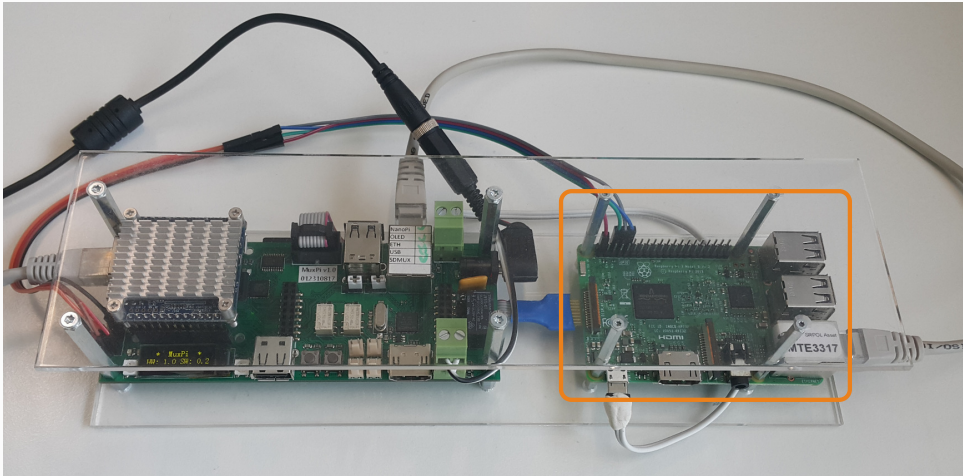
Piece of cake (with MuxPi)



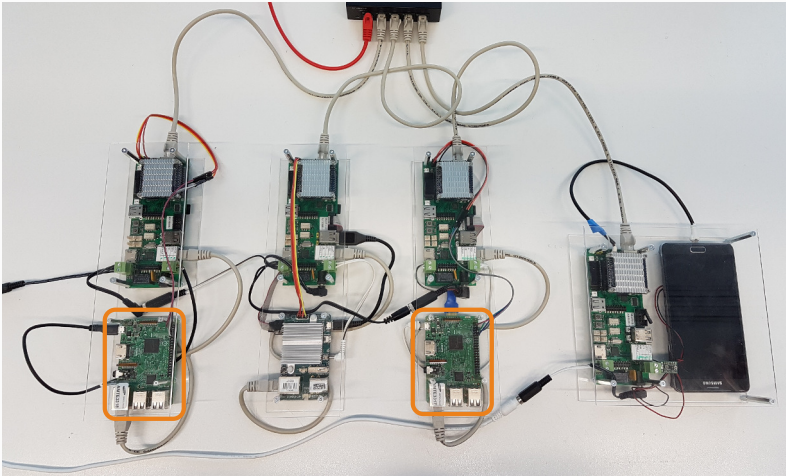
Piece of cake (with Dryad)



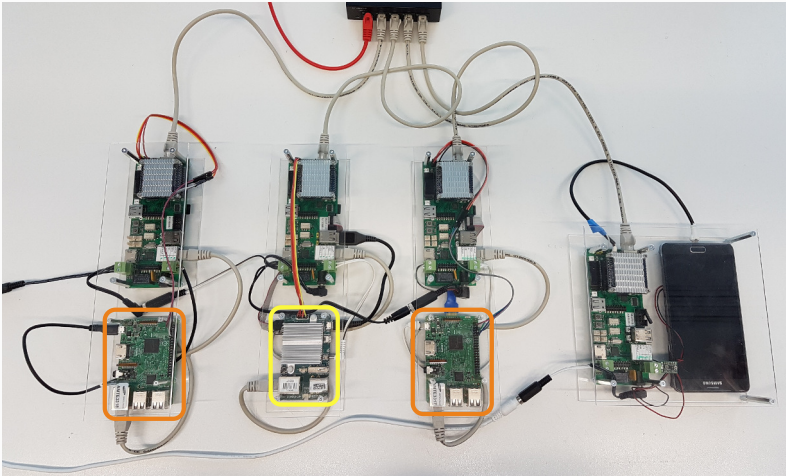
Piece of cake (with Dryad)



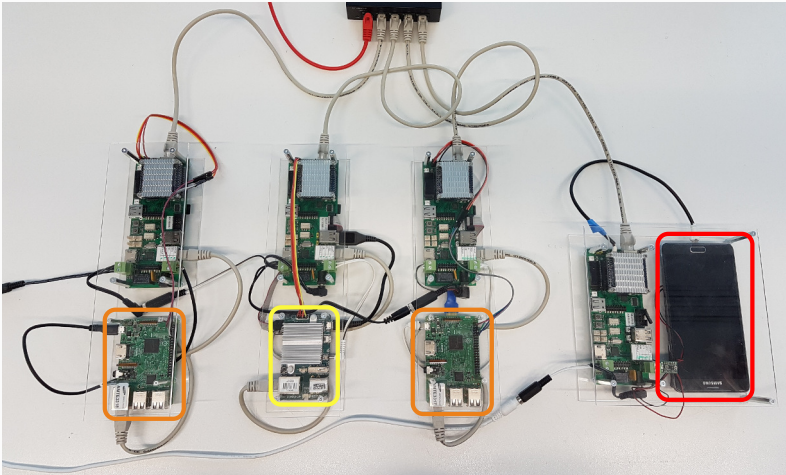
Piece of cake (with Dryads)



Piece of cake (with Dryads)



Piece of cake (with Dryads)



Previous efforts



- Linaro Automated Validation Architecture
- Automation system for deploying operating systems
- Virtual and physical hardware supported
- Allows running boot, bootloader and system level tests

[Home](#)[Jobs](#)[Builds](#)[Boots](#)[SoCs](#)[Compare ^β](#)[Info](#)

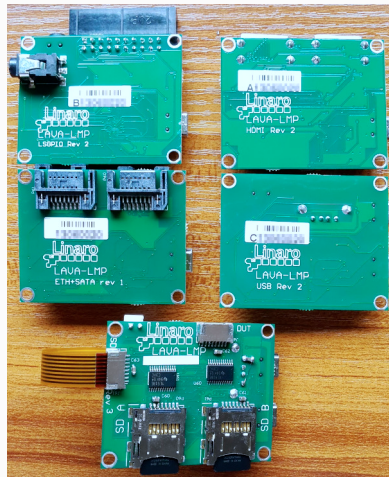
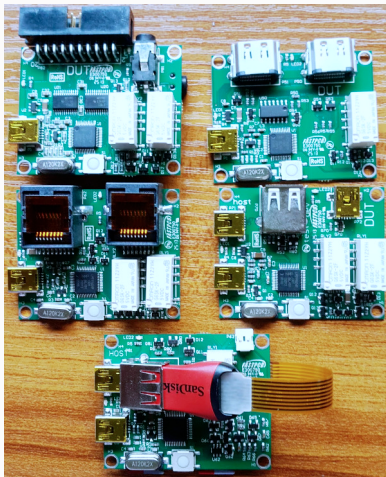
Statistics

Since **May 2014** we:

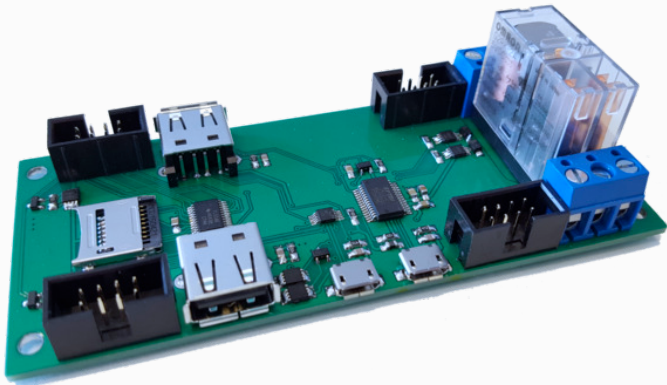
- Ran **13,432** jobs on **65** unique trees and **12,921** unique kernels.
- Performed **2,039,645** builds on **277** unique defconfigs.
- Performed **3,494,550** boots on **271** unique boards, across **3** architectures and **34** unique SoCs.

<https://kernelci.org/stats/>

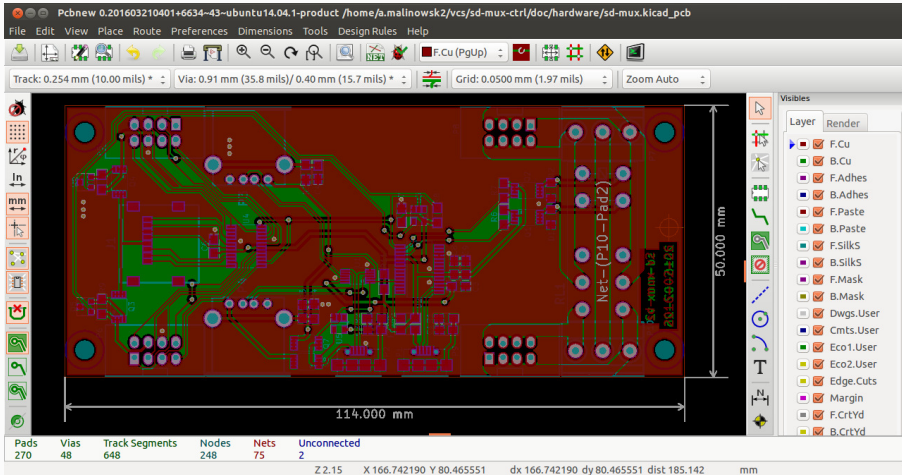
LAVA LMP



<https://linux.codehelp.co.uk/the-problem-of-sd-mux.html>



SD MUX – open hardware



<https://git.tizen.org/cgit/tools/testlab/sd-mux>

Autohat board (SD MUX-based)



<https://github.com/resin-io/autohat-board>

SD MUX issues

```
$ dmesg | tail -12
[ 98.375599] usb 3-1: new full-speed USB device number 12 using xhci_hcd
[ 98.487663] usb 3-1: device descriptor read/64, error -71
[ 98.703656] usb 3-1: device descriptor read/64, error -71
[ 98.919658] usb 3-1: new full-speed USB device number 13 using xhci_hcd
[ 98.919969] usb 3-1: Device not responding to setup address.
[ 99.123998] usb 3-1: Device not responding to setup address.
[ 99.327681] usb 3-1: device not accepting address 13, error -71
[ 99.439718] usb 3-1: new full-speed USB device number 14 using xhci_hcd
[ 99.440049] usb 3-1: Device not responding to setup address.
[ 99.644028] usb 3-1: Device not responding to setup address.
[ 99.847719] usb 3-1: device not accepting address 14, error -71
[ 99.847819] usb usb3-port1: unable to enumerate USB device
```

Idea

- Only **replaceable** media
- No **single point of failure** parts
- No **USB** involvement
(from test server)



Requirements

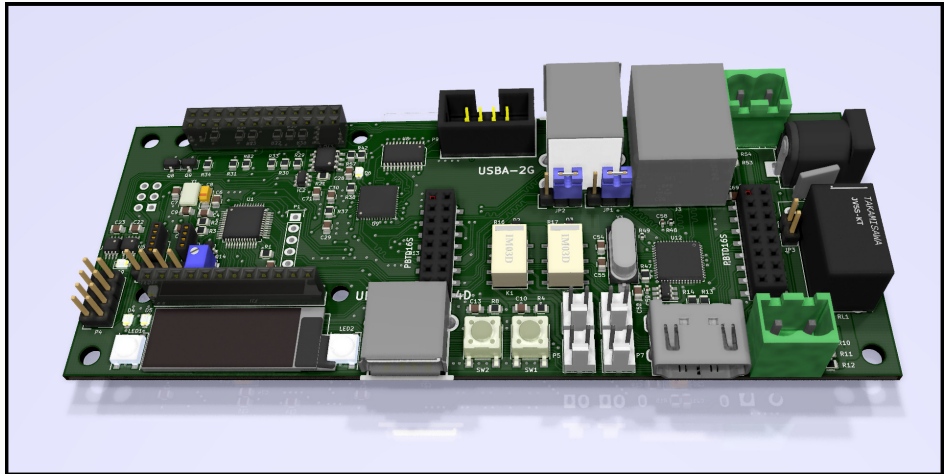
- **Minimum** external connections
- **Unified** remote access to target devices
- **Easy** setup and maintenance



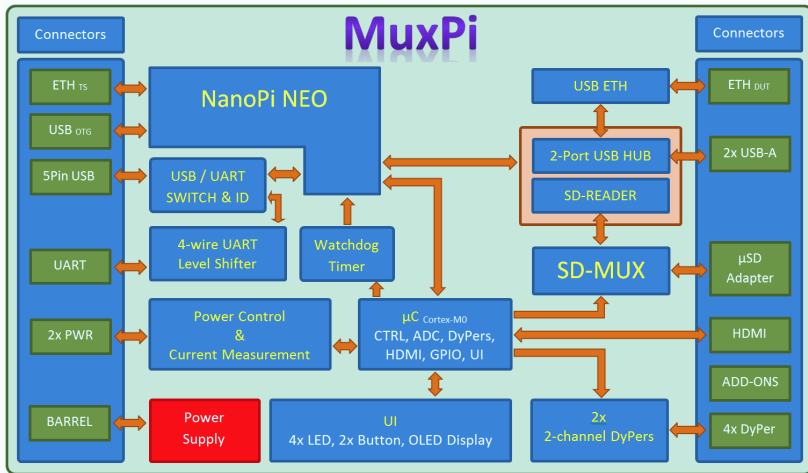
- User interface (often requested)
- Power measurement (increasing demand)
- Writing EDID to HDMI



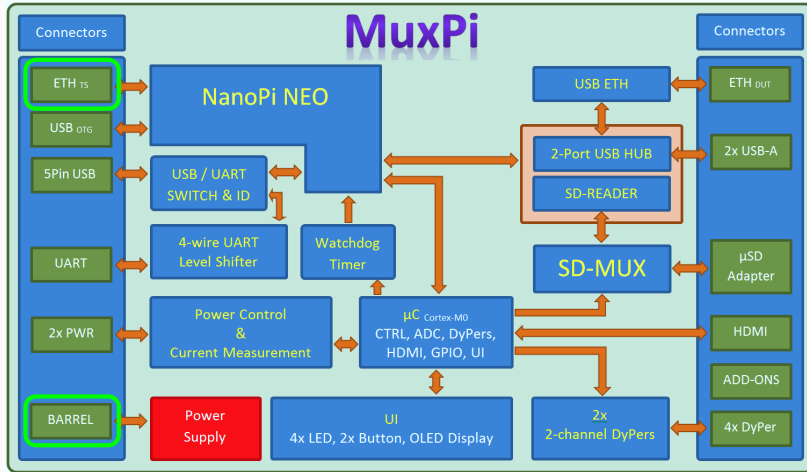
Hardware



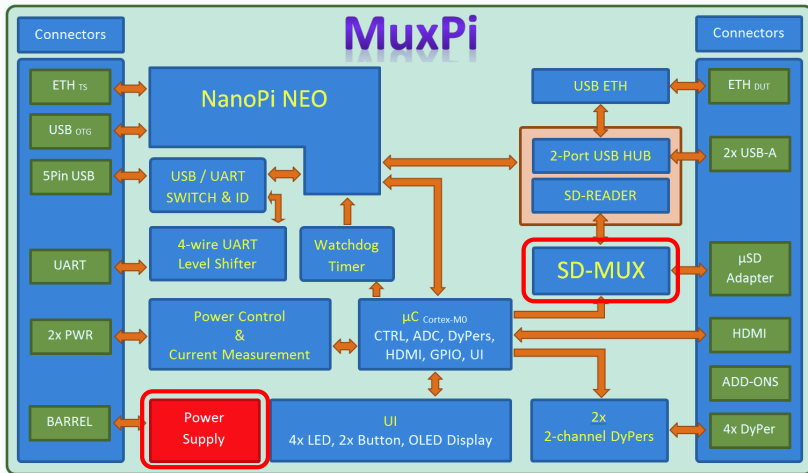
MuxPi components



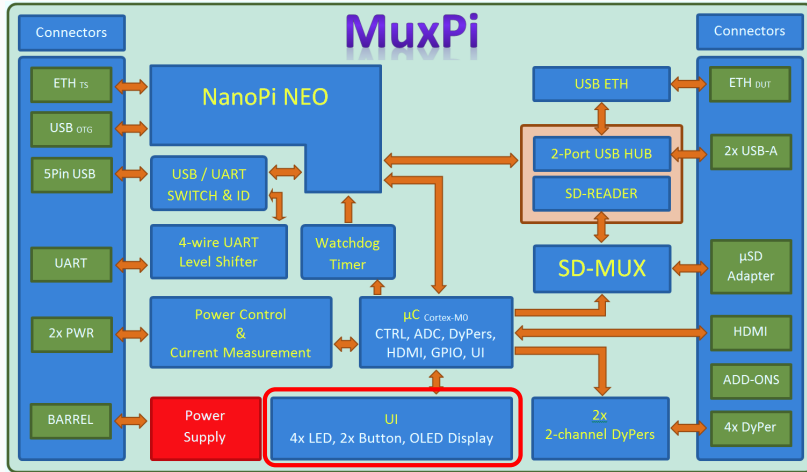
MuxPi components



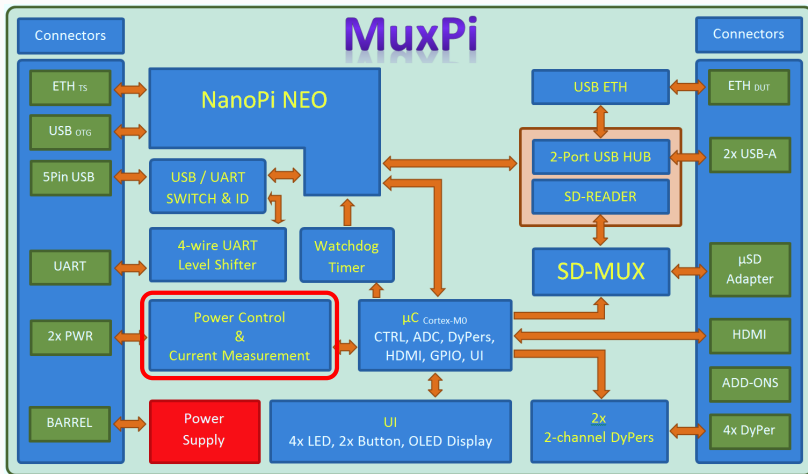
MuxPi components



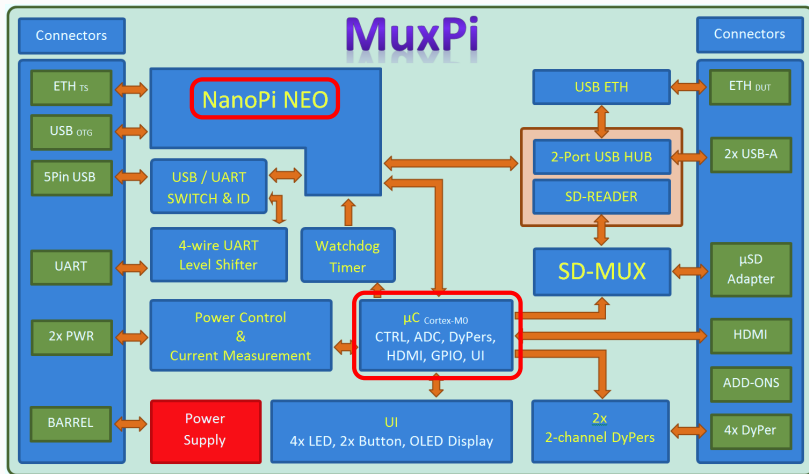
MuxPi components



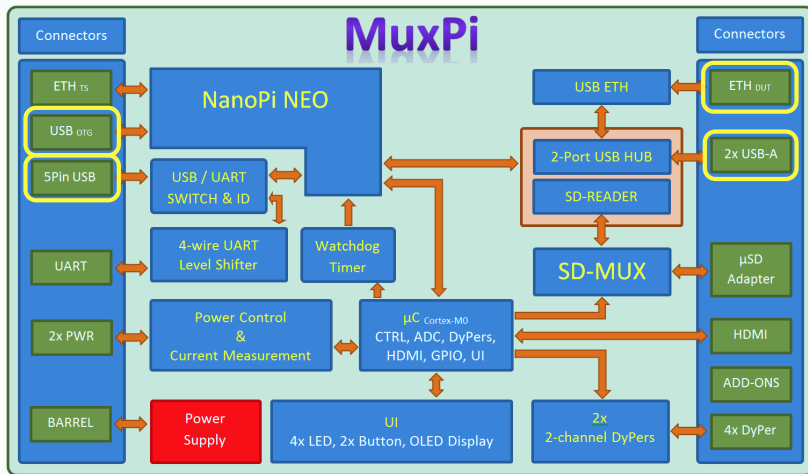
MuxPi components



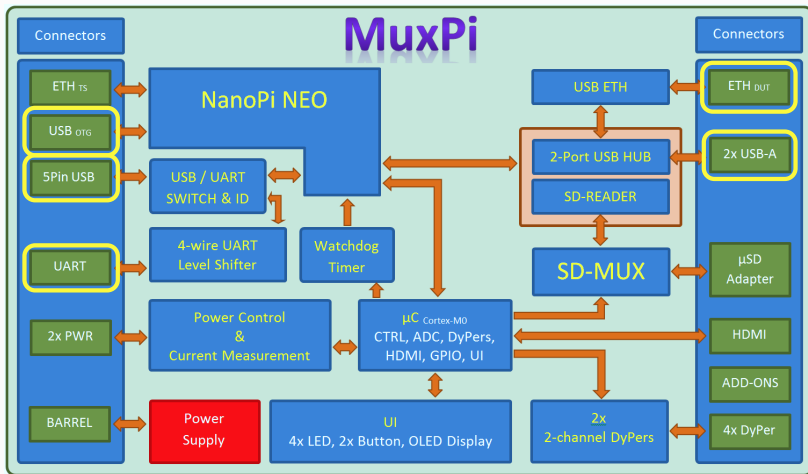
MuxPi components



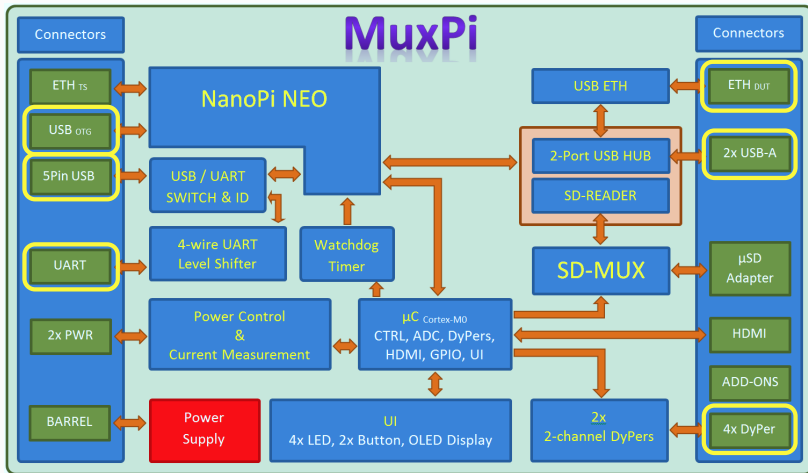
MuxPi components



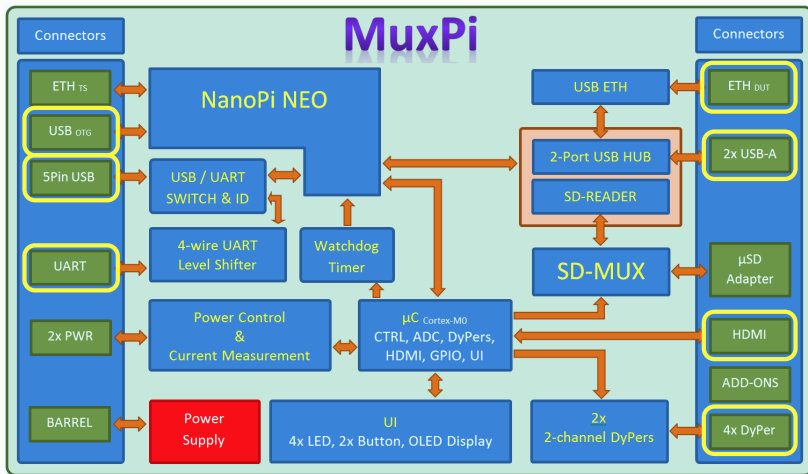
MuxPi components



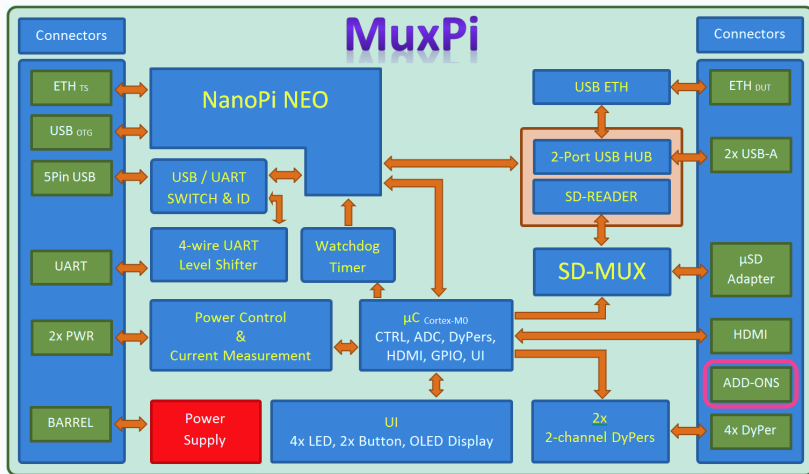
MuxPi components



MuxPi components



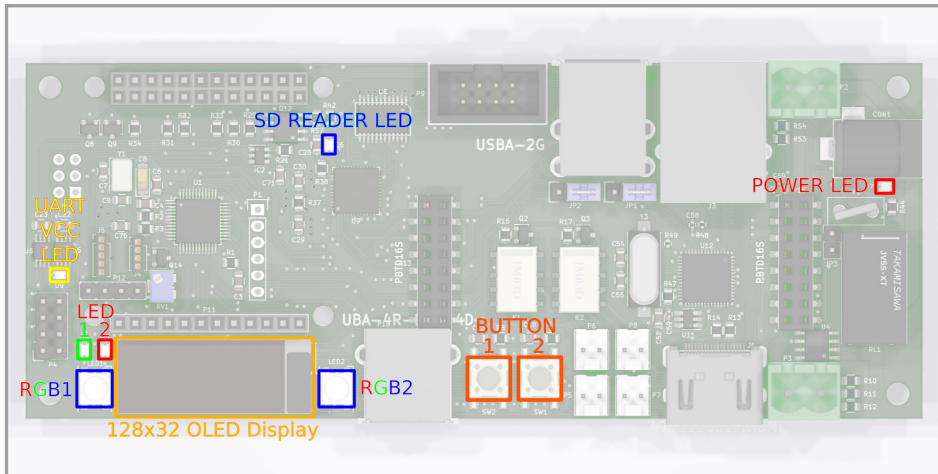
MuxPi components



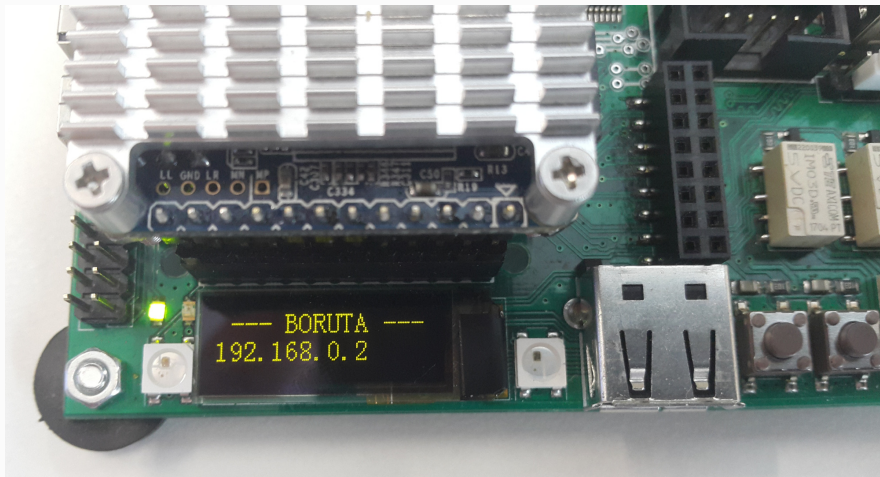
Essential MuxPi functions

- Switching a microSD card between DUT and TS
- Switching power supply for DUT
- Switching jumpers/buttons of DUT
- Measuring power consumption of DUT
- Writing EDID to DUT over HDMI connection
- Providing DUT connection (UART, USB, ETH, microSD card) over Ethernet
- Interacting with farm maintainer

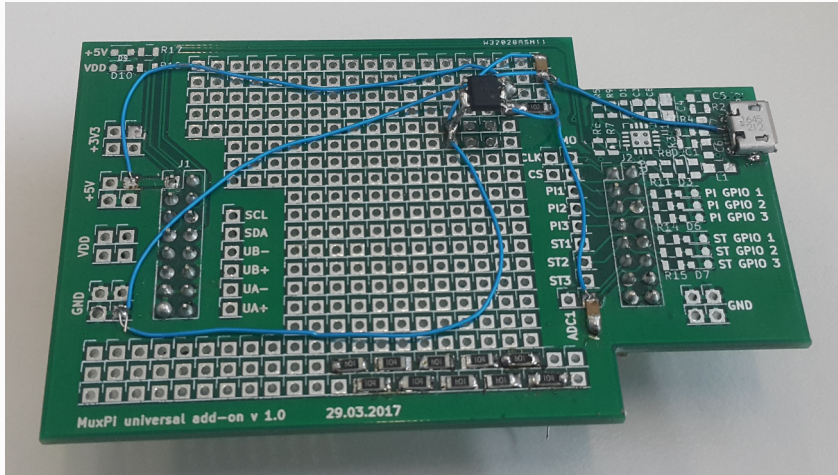
Indicators



Easy maintenance



Extensibility



Major improvements

- **Independent** (standalone)
- **Aware** of its state
- **Easy** to maintain
- **Extensible** from start



NanoPi NEO $\approx$ \$10

NanoPi NEO	≈ \$10
Parts	≈ \$80

Building your own

NanoPi NEO	$\approx$ \$10
Parts	$\approx$ \$80
Soldering skills	<i>High</i>

Building your own

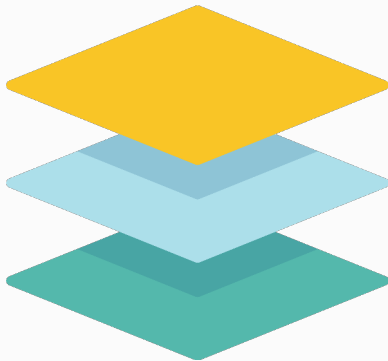
NanoPi NEO	≈ \$10
Parts	≈ \$80
Soldering skills	<i>High</i>
Patience	A LOT

NanoPi NEO	≈ \$10
Parts	≈ \$80
Soldering skills	<i>High</i>
Patience	A LOT

<https://github.com/SamsungSLAV/muxpi>

Software

Multitier architecture



- “Do One Thing and Do It Well”
- RESTful HTTP APIs
- Homogeneous solution stack

Responsibilities



- Who knows what requires verification?
- Who knows which actions are necessary?
- Who knows where can it be done?
- Who knows how to do it?



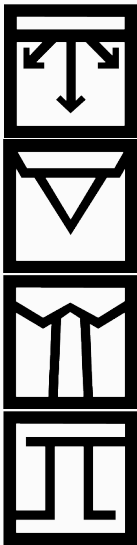
- Who knows what requires verification?
Perun
- Who knows which actions are necessary?
- Who knows where can it be done?
- Who knows how to do it?



- Who knows what requires verification?
Perun
- Who knows which actions are necessary?
Weles
- Who knows where can it be done?
- Who knows how to do it?



- Who knows what requires verification?
Perun
- Who knows which actions are necessary?
Weles
- Who knows where can it be done?
Boruta
- Who knows how to do it?



- Who knows what requires verification?

Perun

- Who knows which actions are necessary?

Weles

- Who knows where can it be done?

Boruta

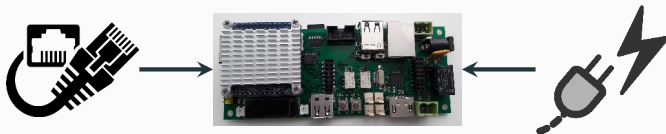
- Who knows how to do it?

Dryad (MuxPi-based)

Dryad (farm)



- **Manages single DUT**
- **Fully aware of its capabilities**
- **Requires only two interfaces**
 - Power supply
 - Network connection (Ethernet)

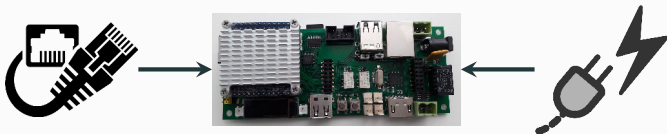


Dryad (software)



```
$ fota --help
Usage of fota:
  -card string
    path to SDcard
  -map string
    path to JSON formatted mapping
  -md5 string
    URL to MD5SUMS file
  -quiet
    suppress logging
```

```
$ stm --help
Usage of stm:
  -dut
    connect SD card to DUT
  -m duration
    time delay for tick command
  -tick
    power off and on after 'm' (s)
  -ts
    connect SD card to test server
```

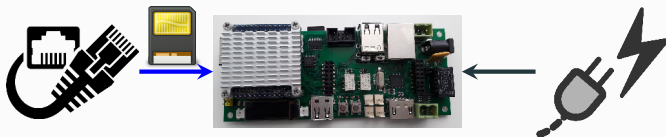


Dryad (software)



```
$ fota --help
Usage of fota:
  -card string
    path to SDcard
  -map string
    path to JSON formatted mapping
  -md5 string
    URL to MD5SUMS file
  -quiet
    suppress logging
```

```
$ stm --help
Usage of stm:
  -dut
    connect SD card to DUT
  -m duration
    time delay for tick command
  -tick
    power off and on after 'm' (s)
  -ts
    connect SD card to test server
```

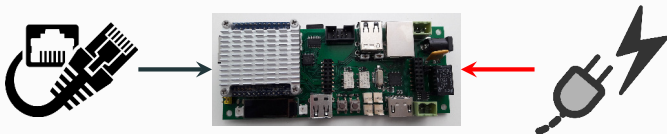


Dryad (software)



```
$ fota --help
Usage of fota:
  -card string
    path to SDcard
  -map string
    path to JSON formatted mapping
  -md5 string
    URL to MD5SUMS file
  -quiet
    suppress logging
```

```
$ stm --help
Usage of stm:
  -dut
    connect SD card to DUT
  -m duration
    time delay for tick command
  -tick
    power off and on after 'm' (s)
  -ts
    connect SD card to test server
```

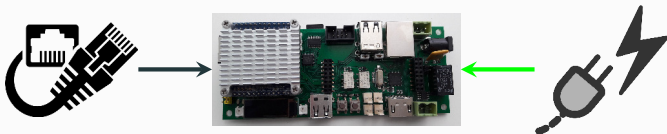


Dryad (software)



```
$ fota --help
Usage of fota:
  -card string
    path to SDcard
  -map string
    path to JSON formatted mapping
  -md5 string
    URL to MD5SUMS file
  -quiet
    suppress logging
```

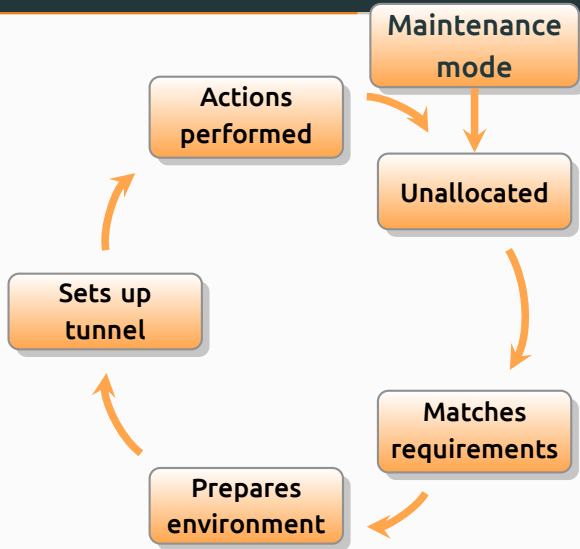
```
$ stm --help
Usage of stm:
  -dut
    connect SD card to DUT
  -m duration
    time delay for tick command
  -tick
    power off and on after 'm' (s)
  -ts
    connect SD card to test server
```

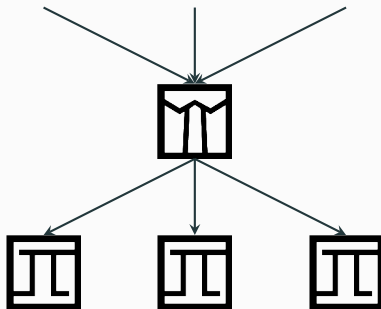


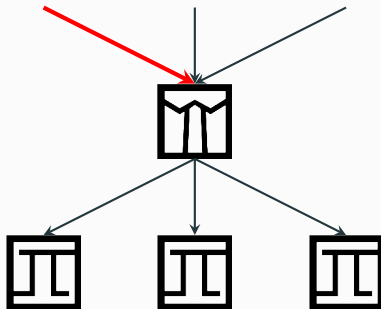


- Dryad farm management system
- Schedules requests
 - Priority
 - Device groups
 - Delayed access
- Provides convenient access to selected Dryad

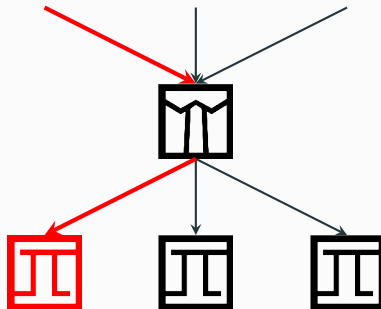
Dryad life cycle in Boruta



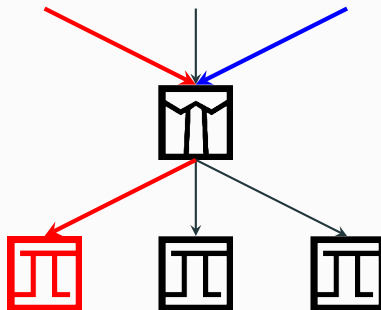




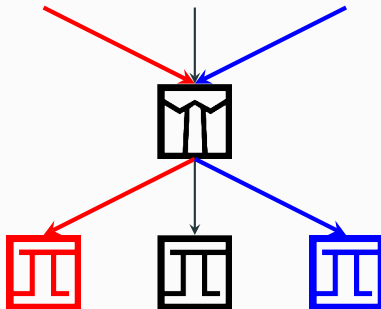
Boruta on stack



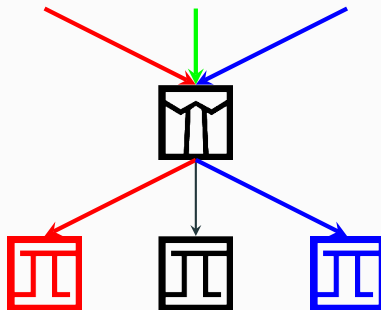
Boruta on stack



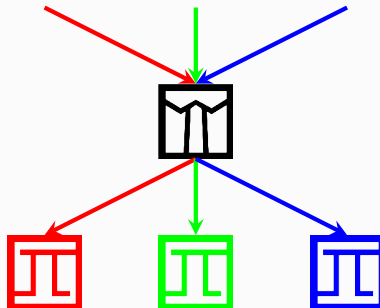
Boruta on stack



Boruta on stack



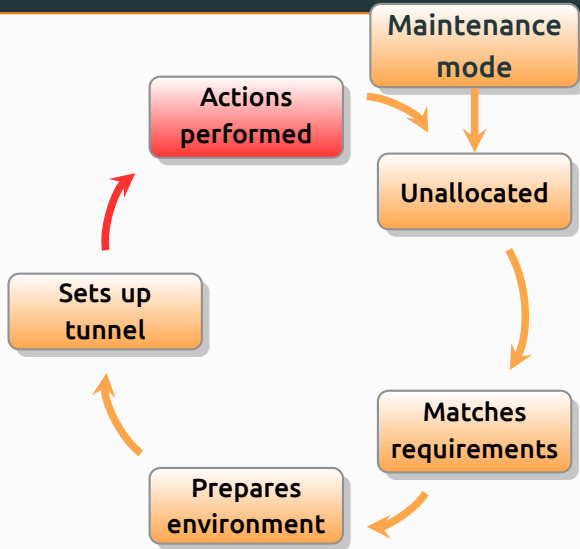
Boruta on stack



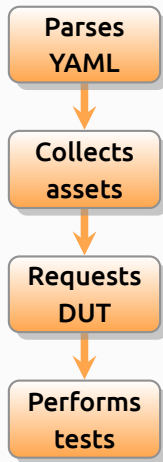


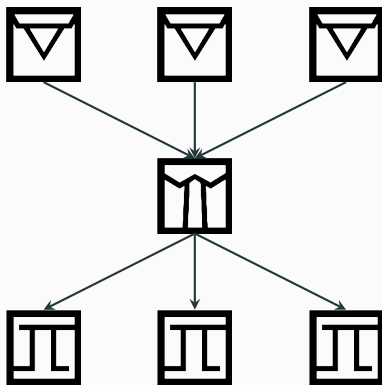
- Lightweight testing framework
- Provides LAVA-like interface
- YAML job definition $\mapsto$ actions executed on DUT
 - Deploy
 - Boot
 - Test
 - Collect

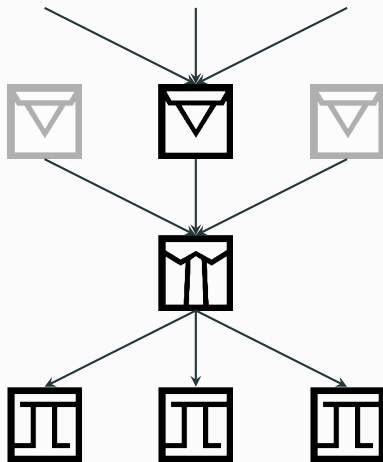
Weles purpose

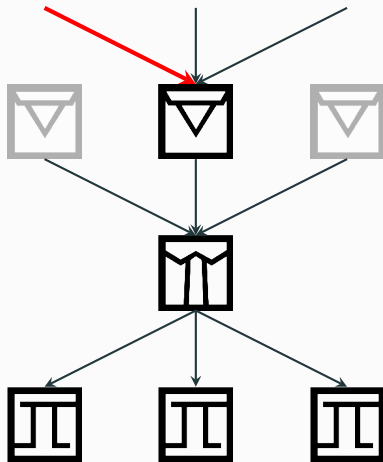


Weles action sequence

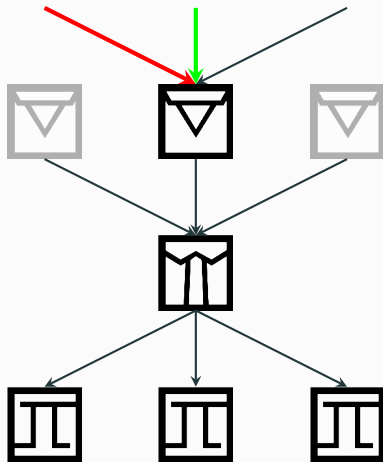




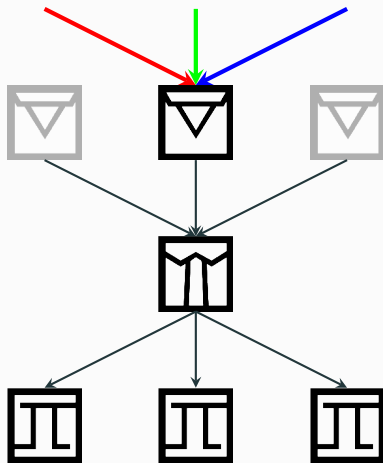




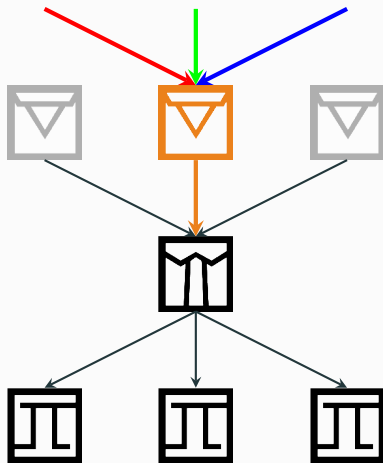
Weles on stack



Weles on stack



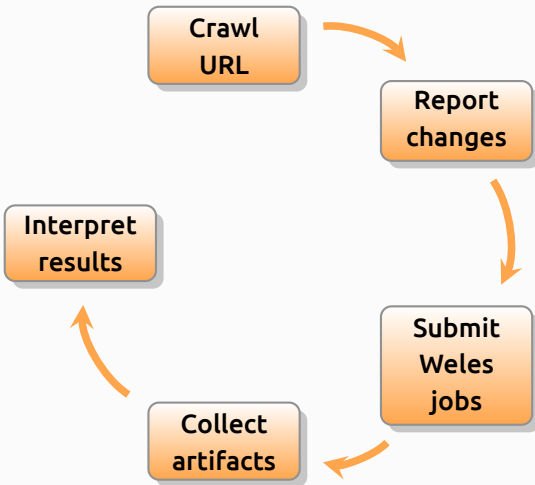
Weles on stack



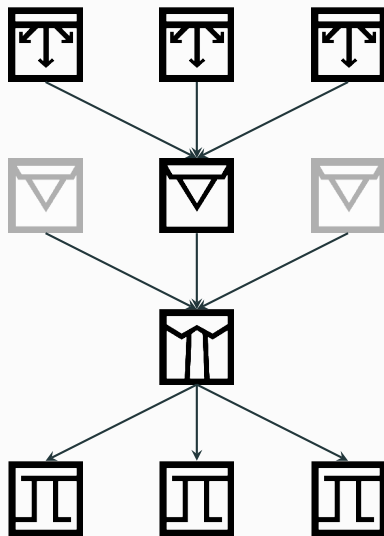


- OS images testing system
- Schedules verification
(per new set of OS images)
- Automates QA step of
Release Engineering Duty

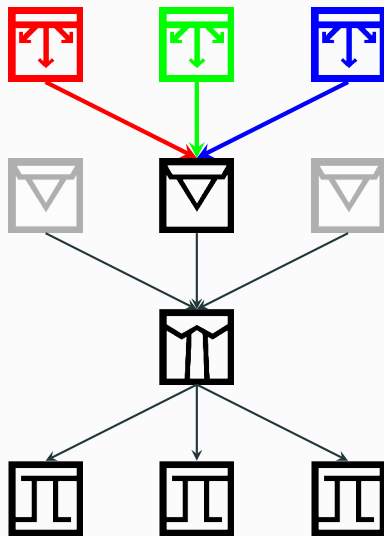
Perun action sequence

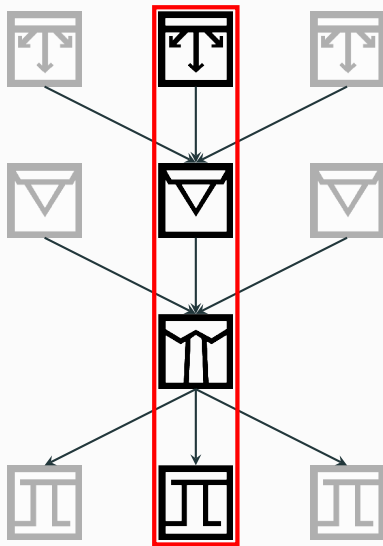
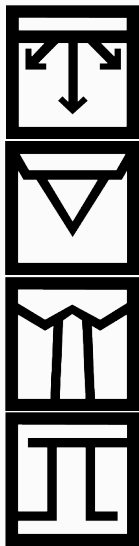


Perun on stack

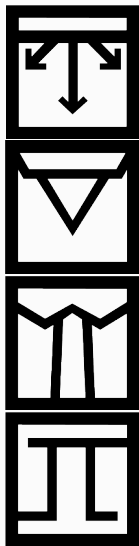


Perun on stack





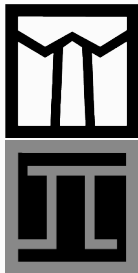
Keeping SLAV simple



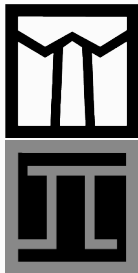
Keeping SLAV simple (and decoupled)



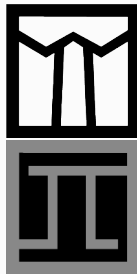
Keeping SLAV simple (and decoupled)



Keeping SLAV simple (and decoupled)



Keeping SLAV simple (and decoupled)



Next steps

Hardware

- Audio I/O
- USB Type C investigation
- NanoPi serial console on USB

Software

- Web interfaces for current layers
- Service state management
- Release engineer's layer

- MuxPi
<https://wiki.tizen.org/MuxPi>
- SD MUX (deprecated – lesson learnt)
https://wiki.tizen.org/SD_MUX

- GitHub

<https://github.com/SamsungSLAV>

- Mailing list

general@lists.tizen.org

Conclusion

- Quick setup
- Easy maintenance
- Responsibilities division
- Execution parallelization
- Environment unification



Questions?

Thank you!

Paweł Wieczorek

p.wieczorek2@samsung.com

Samsung R&D Institute Poland

Acknowledgements

- **Metropolis** – simple, modern Beamer theme

Pictures used

- https://en.wikipedia.org/wiki/File:Heckert_GNU_white.svg
- <https://commons.wikimedia.org/wiki/File:Tux.svg>
- https://commons.wikimedia.org/wiki/File:Wayland_Logo.svg
- https://commons.wikimedia.org/wiki/File:Enlightenment_logo_black.png
- https://developer.tizen.org/sites/default/files/images/about_tizen_1.png
- <https://pixabay.com/en/security-industrial-logistic-1491514/>
- <https://commons.wikimedia.org/wiki/File:ColoredBlankMap-World-10E.svg>
- https://commons.wikimedia.org/wiki/File:Nuvola_Korean_flag.svg
- https://commons.wikimedia.org/wiki/File:Nuvola_Polish_flag.svg
- https://commons.wikimedia.org/wiki/File:Nuvola_USA_flag.svg
- https://validation.linaro.org/static/docs/v2/_images/lava.svg
- https://wiki.linaro.org/Platform/LAB/LMP_in_practice
- <https://forums.resin.io/uploads/resin/original/1X/88ab2e061cd644b18b95fa99ede9ce6b98adfa44.jpg>
- https://commons.wikimedia.org/wiki/File:Italian_traffic_signs_-_fermarsì_e_dare_precedenza_-_stop.svg
- https://farm9.staticflickr.com/8263/28955874330_d1b1202ae8_k_d.jpg
- <https://pixabay.com/en/stars-new-advertisement-sign-146834/>
- <https://pixabay.com/en/update-upgrade-renew-improve-1672351/>
- <https://www.goodfreephotos.com/albums/vector-images/different-colored-layers-vector-file.png>
- <https://pixabay.com/en/ethernet-internet-lan-network-1294340/>
- <https://pixabay.com/en/power-cable-plug-socket-27436/>
- <https://pixabay.com/en/memory-card-sd-secure-digital-98414/>
- <https://pixabay.com/en/kiss-lips-mouth-red-love-rosa-2928081/>
- https://commons.wikimedia.org/wiki/File:PE0-smiley_smile.svg