

# Systems WG – Status and Roadmap

September 7 - 8, 2022 | Virtual Event  
Philipp Ahmann, Robert Bosch GmbH



# Content

- Working group foundation
- The starting activities
- Ongoing activities, roadmap & outlook
- Use case evolution

# “Industrial IoT WG” turned into “**Systems WG**”

- Industrial IoT WG was proposed during April Workshop and kicked off in May as Systems WG
- IoT system architecture is a scaling architecture for many use cases and industries
- Working Group will provide a reproducible reference system
- Strong community support



ELISA Workshop #9  
Is there room for an  
Industrial IoT WG?

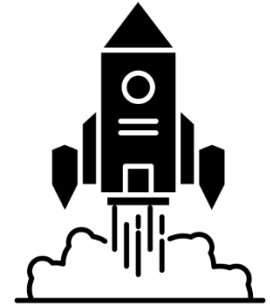
April 5 - 7, 2022  
Philipp Ahmann, Robert Bosch GmbH

Work in Progress - License: CC-BY-4.0

The poster features a blue background with a faint image of a person. On the right side, there is a white hexagonal shield-like shape containing the text 'ELISA WORKSHOPS' and '@PROJECTELISA'. Surrounding this central shape are several circular icons: a group of people, a lightbulb, a car, a heart rate monitor, and a gear. The text 'ELISA' is prominently displayed in a large, stylized font in the background.

# Working Group Goals

- The Systems WG aims to **enable other working groups** within ELISA to put their safety claims towards Linux in a wider system context.
- This is done in the form of a **reproducible reference system based on real-world architectures**, implemented fully based on Open-Source technologies.
- The Systems WG encourages **interactions with other communities and projects**, which either also help **enabling safety use cases with open source software** or plan to make use of mixed-criticality system elements as a base for their product lines.

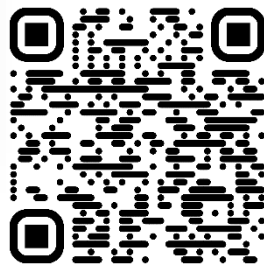


# The start (by Xen Project – Stefano Stabellini)



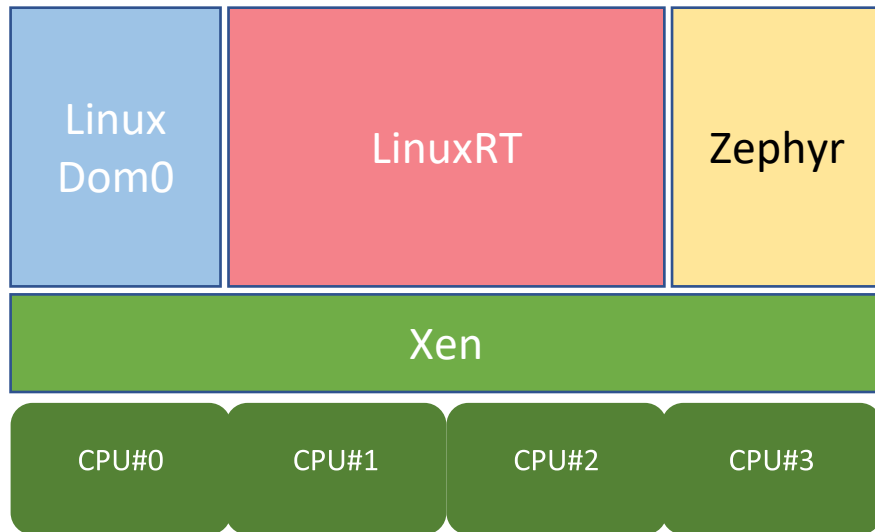
# Static Partitioning with Xen, LinuxRT, and Zephyr: a concrete end-to-end example

Stefano Stabellini  
Embedded Linux Conference 2022



# Content of the Xen end-to-end example

- Build a reference system with default tooling
  - Xen, Linux kernel & rootfs and Zephyr
  - Use ImageBuilder for bootable configuration
  - Xen Device Tree examples
- Give guidance on features (“steps”)
  - Static partitioning
  - Device Assignment
  - Cache Coloring
  - Shared Memory and Event Channels
  - PV drivers



# Some key messages from the presentation

- Some things have been possible for some time, but now are easier
  - **Static partitioning, Device Assignment, Cache Coloring**
  - **Less time required to set up**
  - **Static partitioning required a lot of expertise, now it is automated.**
- Illustrated ability to provide real time deterministic interrupt latency to guests, even under heavy interference
  - **Supported by the system configuration**
- Connects different OSS safety criticality projects in a realistic use case
  - **Taken from a customer example**



# Ongoing activities, roadmap & outlook

# Ongoing Activities

- Porting of Xen end-to-end example to Moulin<sup>1</sup> & yocto
  - Full automation of the end-to-end example (using Moulin)
- Populate ELISA github repository (in alignment with other working groups)
  - Documentation, “glue logic”, configuration
- Find an additional reference system hardware beyond qemu
  - Raspberry Pi and Automotive community reference boards in consideration

## Spin-Off:

- SBOM generation for safety
  - New SPDX “FuSa Special Interest Group” branch

# Full automation of the example (using Moulin)

---

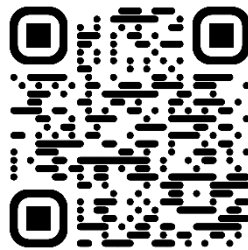
***“Moulin is meta build system that is capable of building multiple & complex images for embedded devices at once.”***

---

- Moulin makes use of “Ninja” build system
- Automation of end-to-end Xen example is WIP with the following features tested and working:
  - Xilinx ZCU102 is supported
  - 3 domains are being built: Linux based Dom0 and DomD, and Zephyr

# New SPDX “FuSa Special Interest Group”

- Draft goal: “***Have a safety case model defined by SPDX tags***”
- Use case:  
Observe changes on main branch and automatically generate a traceability matrix based on the change set for all affected artifacts
- Discussion points are e.g. relationship rules, tags, item types...
- Configuration items list is the base to generate the safety case
- Mailing list & regular call. Join: <https://lists.spdx.org/g/spdx-fusa>
- Meeting minutes: <https://spdx.swinslow.net/p/spdx-fusa-minutes>

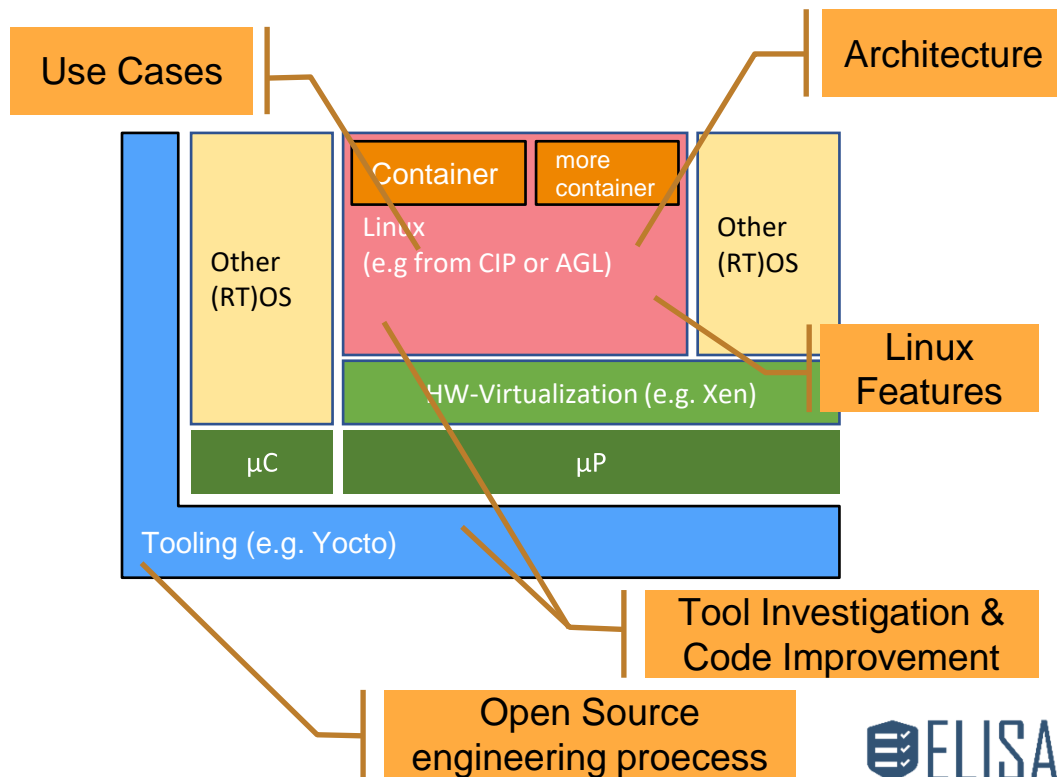


# Next steps (starting after the summit)

- Interact with ELISA working groups
  - Interact with other (safety critical open source) communities
  - Detailed roadmap with work packages and assignable tasks
    - Reproduce end-to-end example with yocto
    - SBOM generation using yocto for systems elements
    - Enable alternative hardware
    - Replace the LinuxRT domain with AGL and/or APERTIS Linux
- ➔ **Break down will be tackled in Manchester**

# Next steps: Interact with ELISA working groups

- **Linux Features, Architecture** and **Code Improvements** should be integrated into the reference system directly
- **Tools** and **Engineering process** should fit the reproducible product creation
- **Medical, Automotive** and future WG use cases should be able to strip down the reference system to their use case demands



# Next steps: Interact with other communities

- Safety-critical open source projects



- Identified OSS project with safety-critical relevance and comparable system architecture considerations



SOAFEE



***"If you have an apple and I have an apple and we exchange these apples then you and I will still each have **one apple**.***

***But if you have an idea and I have an idea and we exchange these ideas, then each of us will have **two ideas**."***

— George Bernard Shaw

# Next steps: Roadmap, work packages, tasks

## REQUEST FOR SUPPORT:

- Reproduce end-to-end example with yocto
- SBOM generation using yocto for systems elements
- Enable alternative hardware
- Replace the LinuxRT domain with AGL and/or APERTIS Linux





# Roadmap outlook *(where do activities lead to)*

## ***Provision of a work bench for (safety-critical) work loads in a reference architecture ecosystem derived from real-world systems***

Targeted activities:		Create flexibility to...
Yocto and moulin automation	→	Integrate the reference image into your product build infrastructure (CI/CD)
AGL and APERTIS Linux	→	Adopt the system to your Linux flavor of choice.
Qemu and embedded HW support	→	Quick start with easy available hardware & fast CI/CD scaling with qemu product
SBOM generation and SPDX SIG	→	Single source, full system SBOM exchange in shared projects
Zephyr & Xen interaction	→	Model real world system architectures

# Use case evolution (scalable architecture)

# Use case evolution

## Think big, start SMALL...

- Today's connected heterogeneous systems share similar system architecture elements
- Starting with a use case with “limited safety criticality” facilitates the path forward to future architectures
- Later applied use cases with higher safety criticality must and will be considered in the initial considerations.







Photo by [Onur Binay](#) on [Unsplash](#)



Photo by [Roberto Nickson](#) on [Unsplash](#)



Photo by [Scott Ymker](#) on [Unsplash](#)



Photo by [Roberto Nickson](#) on [Unsplash](#)



# Join the working group – today!

- Mailing list:  
<https://lists.elisa.tech/g/systems>
- Weekly meeting:  
<https://lists.elisa.tech/g/systems/calendar>
- Meeting minutes  
[https://docs.google.com/document/d/1hHYmTFUG5EypyG-t9RWu1ToBUgEDso42x5fhtdjM\\_bQ](https://docs.google.com/document/d/1hHYmTFUG5EypyG-t9RWu1ToBUgEDso42x5fhtdjM_bQ)
- Repository:  
<https://github.com/elisa-tech/wg-systems>



# Licensing of Workshop Results

All work created during the workshop is licensed under *Creative Commons Attribution 4.0 International (CC-BY-4.0)* [<https://creativecommons.org/licenses/by/4.0/>] by default, or under another suitable open-source license, e.g., GPL-2.0 for kernel code contributions.

## You are free to:

- **Share** — copy and redistribute the material in any medium or format
- **Adapt** — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

## Under the following terms:

**Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

**No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.