How open source projects approach functional safety
Xen, Zephyr & Linux (ELISA)

Nicole Pappler, AlektoMetis
Philipp Ahmann, Robert Bosch GmbH
Credits to: Stefano Stabellini, AMD/Xilinx
Whoami — Philipp Ahmann

- Product Manager for Embedded Open Source
- Chair of the Technical Steering Committee
- Lead of the Systems Working Group
- Member of the Inaugural Advisory Board
- OSS enthusiast and promoter
About Nicole

Professional History:
Been working in production maintenance, automotive, ECU software development
All my projects had some safety criticality
Started to focus on Functional Safety about 12 years ago

Currently:
Tech consulting as part of AlektoMetis
Supporting my customers regarding Functional Safety, Security & compliant use of open source
Involved in some projects:
  - Zephyr (Functional Safety Manager)
  - ELISA (Medical & Systems Group)
  - FuSa for SPDX SIG
  - OpenChain (3rd party certification with TÜV SÜD)

What else?
Not good with remembering names and faces
GitHub, Discord, etc: @nicpappler
What is Functional Safety?

**Definition of Safety**
The freedom from unacceptable risk of physical injury or of damage to the health of people, either directly, or indirectly because of damage to property or the environment.

**Definition of Functional Safety**
The part of safety that depends on a system or equipment operating correctly in response to its inputs. Detecting potentially dangerous conditions, resulting either in the activation of a protective or corrective device or mechanism to prevent hazardous events or in providing mitigation measures to reduce the consequences of the hazardous event.
In Functional Safety you expect:

That the software:

● does behave as specified,
● does not interfere or impair other system components
● and all possible erroneous events are addressed somehow or somewhere.

And you have sufficient evidence to prove this.
Example OSS projects approaching functional safety

Linux:
- ELISA

RTOS:
- Zephyr

Virtualization/Hypervisor:
- Xen
Project Members

Platinum Members
- ANALOG DEVICES
- antmicro
- Google
- intel
- Meta
- NORDIC
- oticon
- Qualcomm Innovation Center
- ZEISS

Silver Members
- AC6
- Arduino
- Baumer
- Baylibre
- blues
- goliath
- Infineon
- irnas
- Laird
- LINARO
- Memfault
- NVIDIA
- Renesas
- Silicon Labs
- STERNUM
- Synopsys
- TII
- Technology Innovation Institute
- Texas Instruments

Premier Members
- Boeing
- Red Hat

General Members
- AISIN
- arm
- ARM
- Bosch
- BOXHINK
- Elektrobit
- HUAWEI
- SUSE
- Suzuki
- WNRVR
Members from Mobility and related industries

No real Mobility or Aerospace member.

Hardware driven:
Mainly Microcontroller and sensor manufacturer.

Mobility supplier.
Originated in server.
Approaching embedded.
No car manufacturer.

(Large non-project member community)

Mobility & Aerospace system provider
Zephyr

- Targeting safety certification from the beginning of the project
- Certification artifacts and safety manual for premium members only
- Safety working group started recently to enable better collaboration
- Naturally, safety awareness in community is limited due to heavy “non-safety” use cases and many unrelated modules.
- Rich ecosystem with strong support for various HW and certain benefits on Linux.
- Posix compatible
Xen

- Since Xen for embedded security working group was started in parallel (in 2010)
- Security & isolation are project’s top priority
- Real-time scheduling.
- Rigorous Quality Process. Full commit traceability.
- Commits are tested with 2 CI loops.
- Widely adopted in critical production environment: (Data center, Desktop & Embedded)
- AMD works on making Xen safety-certifiable
- Continuous certification in mind.
- Phase 1: Certification Concept Approval
- Phase 2: Final Assessment.
Linux

- Open source software superlative.
- Largest community, largest source base.
- Made for flexibility and wide use cases.
- Spread over whole world and in space.
- Several attempts with certification path.
- Gains again momentum for high performance products (e.g. SDV*)
- Prominent open space examples: SIL2LinuxMP and ELISA

*SDV: Software-Defined-Vehicle
Limitations! The OSS projects collaboration …

- cannot engineer your system to be safe.
- cannot ensure that you know how to apply the described processes and methods.
- cannot create an out-of-tree system for safety-critical applications. (continuous process improvement argument!)
- cannot relieve you from your responsibilities, legal obligations and liabilities.

But…

Projects provide a path forward and peers to collaborate with!
Certification financing

Platinum members

AMD/Xilinx

Integrators (like RedHat)
Fully open vs. Pretty open

<table>
<thead>
<tr>
<th>Recently started safety-wg for better collaboration.</th>
<th>Discussions are open (to participate you need to have a copy of “Misra-C”)</th>
<th>Completely open to everyone.</th>
</tr>
</thead>
<tbody>
<tr>
<td>New life to activity due to openness.</td>
<td>Misra-C, documentation and other parts are open source and upstream.</td>
<td>Focus is on tools, kernel improvements, documentation and processes.</td>
</tr>
<tr>
<td>Example: requirements tool (StrictDoc)</td>
<td>Safety manual and other safety artifacts will be made commercially available via AMD/Xilinx</td>
<td>Outcome enables other integrators to build their products around Linux.</td>
</tr>
<tr>
<td>Some results remain “behind the scenes” for premium members</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Due to smaller (upstream) code size, it can be easier to certify Xen or Zephyr. Also, complexity/features may be decreased/stripped (no L2 caches or dynamic memory allocation).
Trainings

Provide(d) IEC 61508 training by TÜV SÜD for project members (some contributors/maintainers have official safety training)

The safety committee (and safety working group) mainly consist of experienced safety experts.

Misra-C trainings for project contributors via Bugseng sponsored by AMD.

Mainly 1 safety expert, many engineers with safety in mind and practical product experience

Special topic webinars within ELISA.

No direct ISO26262 or IEC61508 trainings for ELISA members.

Many experienced safety experts within ELISA project.
Challenges: Linux in safety critical systems

The Linux kernel has:

• Large Development Ecosystem
• Security Capabilities
• Multi-Core Support
• Unmatched Hardware Support
• Many Linux Experts at all levels available

Traditional safety-critical OS has:

• Hard Real-time Capabilities
• Proven Safety-compliant Development Process
• …

Can these differences be tackled?
ELISA Working Groups - Fit in an exemplary system

- **Linux Features, Architecture** and **Code Improvements** should be integrated into the reference system directly.

- **Tools** and **Engineering process** should serve the reproducible product creation.

- **Medical, Automotive, Aerospace** and future WG use cases should be able to strip down the reference system to their use case demands.
The mission of the project is to define and maintain a common set of elements, processes and tools that can be incorporated into Linux-based, safety-critical systems amenable to safety certification.”

The scope of the project includes software and documentation development under an OSI-approved license supporting the mission, including documentation, testing, integration and the creation of other artifacts that aid the development, deployment, operation or adoption of the project.”

from the technical charter
Safety Critical Systems

“Assessing whether a system is safe, requires understanding the system sufficiently.”

• Understand your system element within that system context and how it is used in that system.

• Select system components and features that can be evaluated for safety.

• Identify gaps that exist where more work is needed to evaluate safety sufficiently.
Safety Element out of Context (SEooC)

Element that can prove it has sufficient evidence,
- can be integrated to a safety relevant system,
- target system is unknown during the SEooCs development.

Actually: Element of assumed context
- Provides a product with safety critical properties within a defined (functional) scope
- Provides information how it needs to be integrated
- Ships with a Safety Manual

Obligations!
- Scope and capabilities of the SEooC must match with the final system’s safety needs!
- If the system safety is insufficient, a safety SEooC will not safe you!
- Adherence to the Safety Manual!
Community challenges for all projects

Bring the argument of „OSS is not behaving like commercial software“. Less influence on maintainers (positive & negative – no traditional supplier management).

Harder to train/direct developers (but some Xen community members got Misra-C trainings and Zephyr members IEC 61508 trainings).

Liability of a community? (but commercial provider may be liable – insurance)

Development process: Requirements, traceability, v-model,… mapping safety integrity standards
Interactions between the communities

- Open source projects focusing on safety-critical analysis
  - Xen Project
  - Zephyr

- Open source projects with safety-critical relevance and comparable system architecture considerations
  - Automotive Grade Linux
  - SOAFE
  - SDV

- Further community interactions
  - Yocto Project
  - SPDX
  - Linaro

― George Bernard Shaw

“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
Zephyr – Compliant Development: V-Model

It is a challenge to map a stereotypical open-source development to the V-model

- Specification of features
- Comprehensive documentation
- Traceability from requirements to source code
- Number of committers, commits and information known
- Test coverage metrics

Provide the evidence that open source developed items can map to compliance and meet all requirements
What’s happening in Zephyr now…

**Safety Committee**
- Safety Certification strategy decisions
  - scope of certification
  - certification standards
  - certification timeline
- Assessment and audit specific tasks
- Owner of certification artefacts
- Participation limited to the project’s platinum members, the safety architect and the functional safety manager

**Safety Working Group**
- Enabling safety qualifications/certifications in the project
- Working on created the required documentation and evidences
  - Setting up requirements management tooling
  - Creating/deriving and documenting requirements
- Open to everyone to participate
Snapshot: Current Requirements Work

- Used tooling: StrictDoc (https://github.com/strictdoc-project/strictdoc)
- Decision on UIDs for requirements (will be generated by StrictDoc)
- Hierarchical structure of requirements that works for the project
- WIP: capturing requirements in StrictDoc
Join the talk on Sunday

Application of the SPDX Safety Profile in the Safety Scope of the Zephyr Project

Recommendations for new contributors

● Just show up – All presented projects are open for the adaptation of new use cases, input, domain-specific working groups etc.

● Share Safety Best Practice: Functional and structural expectations of the component used in the context of the entire system

● Become an OSS evangelist: Open source can already be used in a variety of safety contexts. Knowledge of the actual structure and potential is very scarce in the field of assessors, notified bodies and related authorities.
Bet on certification (if and when)!!
Thank you.
Getting involved with ELISA

https://elisa.tech

https://github.com/elisa-tech

https://lists.elisa.tech

https://www.youtube.com/@elisaproject8453
Getting involved with Zephyr

https://www.zephyrproject.org

https://www.github.com/zephyrproject

https://lists.zephyrproject.org

https://chat.zephyrproject.org
Getting involved with Xen

https://www.xenproject.org

https://github.com/xen-project

https://xenproject.org/help/mailing-list/

https://xenproject.org/help/matrix/