



Enabling Linux in Safety Applications

The Enabling Linux In Safety Applications (ELISA) project aims to make it easier for companies to build and certify Linux-based safety-critical applications - systems whose failure could result in loss of human life, significant property damage or environmental damage.

ELISA members are working together to define and maintain a common set of tools and processes that can help companies demonstrate that a specific Linux-based system meets the necessary safety requirements for certification. ELISA is also working with certification authorities and standardization bodies in multiple industries to establish how Linux can be used as a component in safety-critical systems.

WHAT YOU CAN USE TODAY

BASIL: Requirements, tests and traceability.

stress-ng: Linux hardware and software stress testing.

Reference architectures: Published designs and the external watchdog pattern.

ELISA Directory: A common entry point to artifacts, deliverables, working groups and technical resources.

Working-group reports and meeting minutes: Public resources to explore.

WORKING GROUPS & SPECIAL INTEREST GROUPS

Working Groups

- Aerospace
- Automotive
- Open Source Engineering Process
- Linux Features for Safety-Critical Systems
- Medical Devices
- Safety Architecture
- Systems
- Tools

Special Interest Groups

- Space Grade Linux
- Railways
- Lighthouse OSS

Each Working Group and Special Interest Group hosts recurring meetings that are open to the community, with recordings and updates shared through project mailing lists.

MEMBER ORGANIZATIONS

PREMIER



GENERAL



ASSOCIATE



Get Involved: <https://elisa.tech/>



Building the safety argument

Functional safety • Linux kernel • Open collaboration

WHY LINUX, WHY NOW?

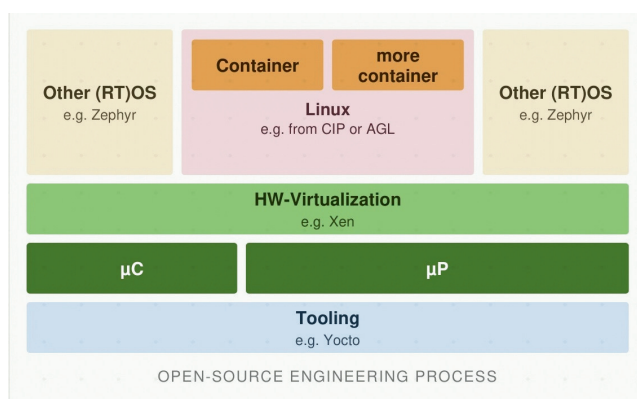
Linux already runs in cars, medical devices and spacecraft, covering user-facing applications, telemetry, control and non-safety compute. What is missing is a documented path from Linux kernel behavior to a complete safety argument. ELISA builds that path collaboratively, reducing duplicated effort and creating reusable knowledge for manufacturers, suppliers, integrators and distributors.

TECHNICAL SCOPE

ELISA develops artifacts that support safety cases; it does not certify software itself.

- System and kernel analysis processes
- Safety-relevant kernel features
- Argumentation for IEC 61508 / ISO 26262 objectives
- Reproducible reference architectures

EXEMPLARY REFERENCE ARCHITECTURE



REFERENCE: AUTOMOTIVE

Information presented to a vehicle driver through warning indicators is often safety critical, for example oil temperature, engine diagnostics, or gear status. Such information must not fail silently. The reference design renders these tell-tales in a visual instrument cluster. An external challenge-response watchdog independently confirms that the rendering loop is alive, and a fallback path takes over if it is not. Both the architecture and the watchdog sample implementation are published by the ELISA project.



REFERENCE: AEROSPACE

The ELISA Aerospace Working Group is taking its first steps toward sharing how Linux is used in flight; it builds on general safety reference concepts applied to the avionics domain. The main shift is the target of the safety argument from the automotive functional-safety standards to DO-178C design assurance. Building on that pattern, the working group researches the capabilities a flight-worthy Linux needs: application monitoring and runtime verification to confirm workloads meet their timing and behavioral contracts, and kernel code minimization to shrink and characterize the certifiable footprint. The same scope boundaries hold as in the automotive exemplar: ELISA develops artifacts that support safety cases rather than certifying software, and its work targets concepts to aid in certification efforts.

✓ Safety ≠ Security

ELISA's scope is functional safety — predictable, fail-safe behavior under fault, not protection against a malicious actor. The two disciplines overlap (a compromised kernel can't be argued safe), but ELISA's artifacts assume a non-adversarial fault model. Security hardening (e.g. kernel CVE response, secure boot) is complementary, out of scope, and covered by other Linux Foundation projects such as the Linux kernel's own security team.

Get Involved: <https://elisa.tech/>