ELISA

Enabling Linux in Safety Applications

Making Penguins Fly

Presenters

Matthew Weber

Associate Technical Fellow @
The Boeing Company

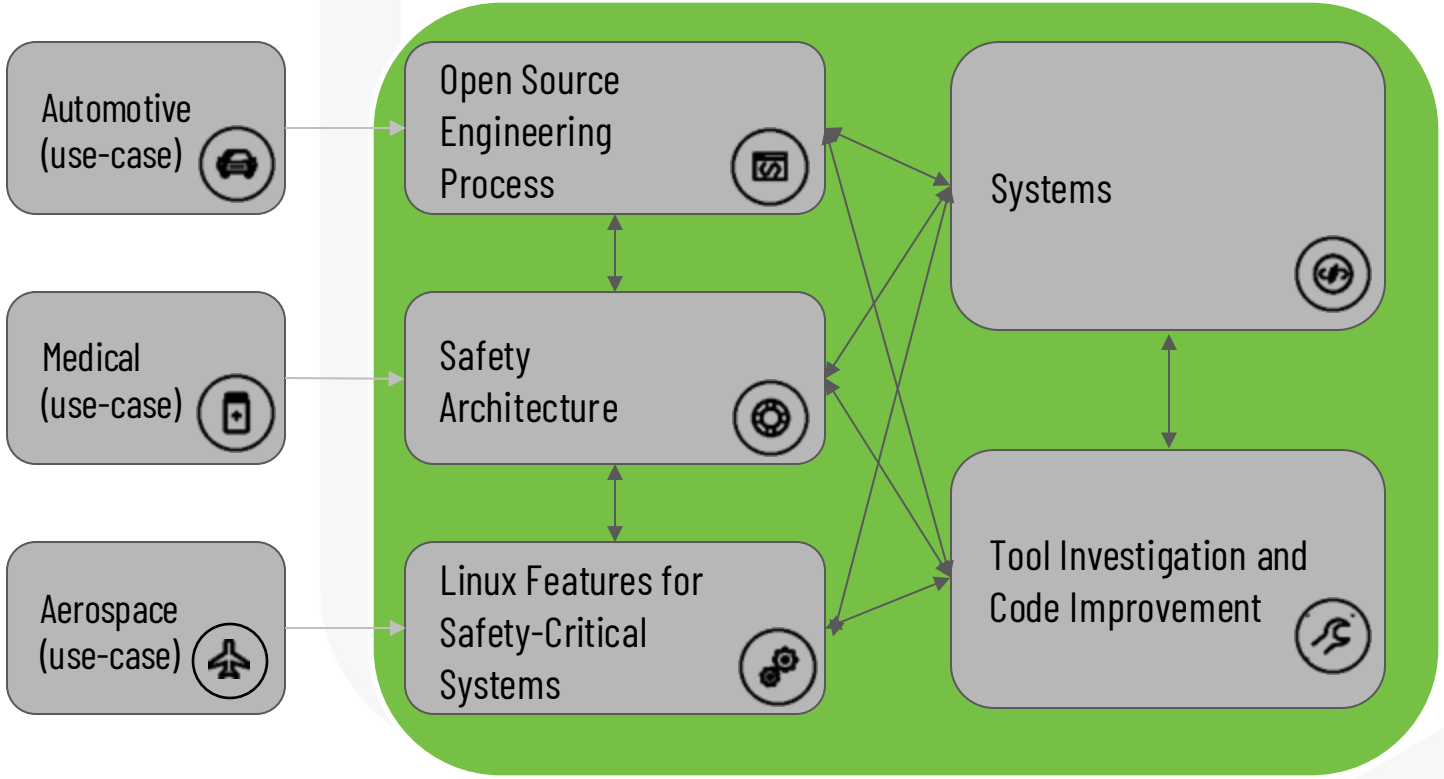
Michael Monaghan

Software Engineer @
NASA Goddard Space Fight Center

ELISA
Enabling Linux in Safety Applications

ADVANCING LINUX IN SAFETY-CRITICAL SYSTEMS

… aiming to make it easier for companies to build and certify Linux-based safety-critical systems whose failure could result in loss of human life, significant property damage or environmental damage.



ELISA
Enabling Linux in Safety Applications

Aerospace Working Group

“… shall develop use cases to inform and influence Linux architecture and related tools, work to derive technical requirements for avionics operating systems, and seek to enhance and expand avionics software lifecycle processes, practices, and tools to enable use of Linux in avionics systems that are certified to high design assurance levels.”

See our annual briefing for more 2024 details and plans for 2025

(Slides)
(Recordings)

ELISA
Enabling Linux in Safety Applications

NASA Goddard Workshop

Hosted by our friends at NASA, we hosted the first in person meeting for the group with great success.

- Two day event
- Attendees: 30 in-person & 40 virtual
- 18 Talks & 20 Speakers
- Tour of NASA Facilities
- Some of the attending organizations: Red Hat, Bosch, NASA, Wind River, TelePIX, Sony, Linux Foundation

Link to workshop's content and video recordings

ELISA
Enabling Linux in Safety Applications

NASA Goddard Workshop: Sessions

Title	Presenter(s)
Welcomes + Orientation	Michael, Philipp, Kate, Ramon
Space Grade Linux Intro	Michael Monaghan - NASA
Lessons from Automotive Grade Linux	Walt Miner - Linux Foundation
Linking external Test Results to Test Cases in BASIL to support pre existing test infrastructures	Luigi Pellecchia - Red Hat
How to use ks-nav for a feasible and meaningful test campaign in the Kernel	Alessandro Carminati - Red Hat
Verification and Validation of the OS and "certification package"	Scott Tashakkor - NASA
Test and Assurance of Non-Volatile Memory Devices for Space	Ted Wilcox - NASA
Linux Kernel Design Documentation	Gabriele Paoloni - Red Hat, Chuck Wolber, Kate Stewart - Linux Foundation
Space Grade Linux interest survey results	Kate Stewart, Ramon Roche - Linux Foundation
F prime	Michael Starch - NASA
Space ROS	Ivan Perez - NASA
cFS Overview Presentation	Richard Landau, Ashok Prajapati (NASA)
Investigating the Implementation of Linux-based Payload Computers: A Review of In-Orbit Demonstrations for Edge AI in Space Missions.	Dongshik Won - TelePIX Co., Ltd.
Container and immutable patterns for operating systems and workloads	Michael Epley, Tony James - Red Hat
Containerization in Space: Podman for Mission-Critical Operations and Resilience	Dan Walsh, Douglas Schilling Landgraf -Red Hat
Real Time Linux update	Steven Rostedt - Google
Building an OSS Ecosystem for Space	Tim Bird - Sony

Link to workshop's content and video recordings

ELISA
Enabling Linux in Safety Applications

Where to start?

Join our call on the 3rd Thursday of each Month

Register here to receive a calendar invite

https://elisa.tech/community/meetings/

ELISA
Enabling Linux in Safety Applications

Space Grade Linux

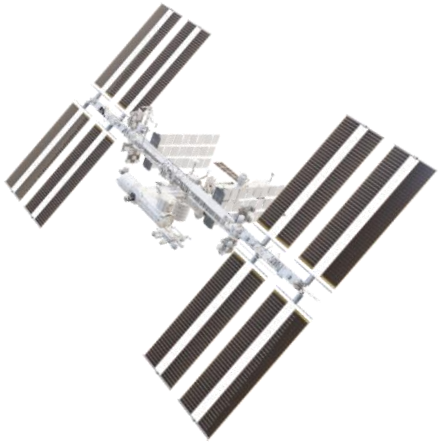
FSW25 Introduction - Michael Monaghan, NASA Goddard Space Flight Center

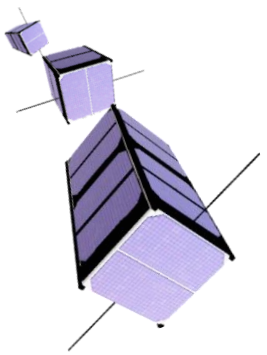
Why use Linux?

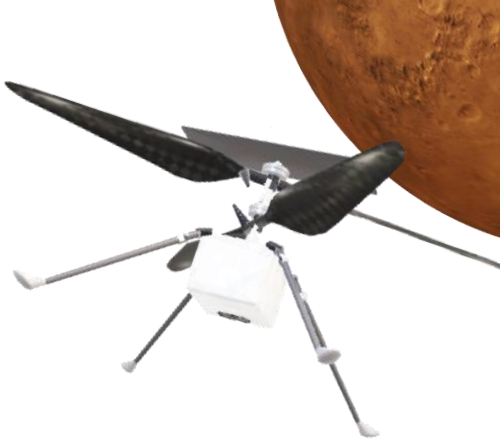
Space Industry is adopting Linux, **Fast!**

- Free and open source
- Abundant workforce expertise
- Unparalleled software ecosystem
- Strong embedded hardware support
- Extensively reviewed and tested
- Adaptable to real-time or performance-centric workloads

Who is using Linux?







Space

- NASA
  - Ingenuity
  - International Space Station Experiments
  - Starling
  - Countless CubeSats
- SpaceX
  - Falcon 9
  - Dragon
  - Starlink (>30k Linux nodes!)

Earth

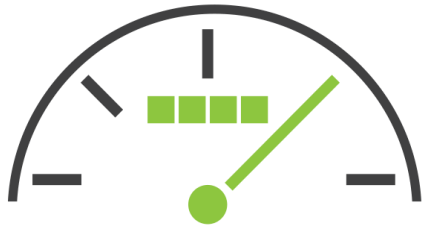
- Automotive
  - Major Autopilot/Advanced driver assistance platforms
- Telecom
  - Majority of LTE and 5G chipsets.
  - LTE and 5G infrastructure

…

What is SGL?

Space Grade Linux is a collaborative, open-source project bringing together space agencies, industry, and academia to establish a trusted ecosystem of Linux-based, open-source software for mission critical spacecraft operations.

Inspiration



AUTOMOTIVE GRADE LINUX

- Revolutionized automotive software.
- Brought competitors together to collaborate on common code.
- Quality of life improvements beyond the automotive industry.

Purpose of SGL

Reduce barrier to entry, cost, and error.

- Stop reinventing the wheel!
- Reuse software and collaborate.
- Minimize time to launch.
- Condense required expertise.
- Tap into collective knowledge.

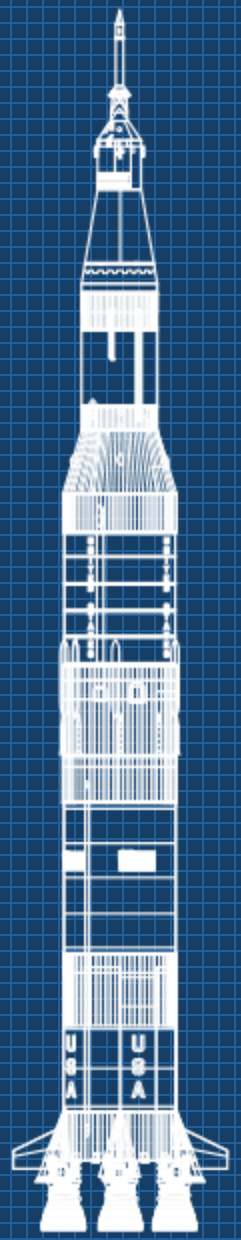
Facilitate modern workloads in space

- Prepare for next-gen high performance spacecraft processors.
- Tap into unrivaled performance and software support offered by Linux
- Support nearly all AI/ML frameworks out of the box.
- Address challenges presented by spacecraft environments.

Inspire Confidence in Linux

- Establish a trusted ecosystem.
- Demonstrate reliability outside terrestrial applications.
- Serve as a baseline for future certification.

SGL Blueprint



Your Mission (Software Payload)

- *Mission Specific Flight Software (cFS Apps, FPGA design, drivers…)*

Space Grade Linux

- Userspace
  - Software Framework (e.g. cFS)
  - Linux-specific framework apps.
  - Daemons and tools (systemd, udev…)
  - Dependencies and libraries (busybox, glibc…)
  - Dependency and library config
  - Distribution policy and config
- Board Support Package
  - Kernel & Config
  - Drivers
  - Bootloader (e.g. u-boot)
  - Bootloader additions (resilient boot flow, update mechanisms)
  - Hardware Platform Support
- Baseline Certification
  - Kernel + Userspace
  - Security hardening

How?

Yocto Project® / OpenEmbedded

- Build system and interface to contribute.
  - Custom kernels, bootloaders, board support packages, software packages, distribution configurations, etc.
- Vast preexisting software and hardware support.
  - layers.openembedded.org
- Widely used across industry.

Focus Areas

Software

Yocto Linux Disto tailored to space.

Stop re-inventing the wheel.

Reduce time to launch.

Establish a common code base.

Support cutting edge open-source software, such as for AI/ML.

Hardware

Yocto BSP for common space processors.

Facilitate high-performance mission processing needs.

Target next-gen space processors.

Meet aggressive SWaP constraints.

Safety

High design assurance levels.

Agency certifiable for Mission Critical Ops.

Establish a track record.

System security and integrity by design.

Protection against malicious actors.

Status

Planning phase

- Discussions
- Knowledge sharing

Initial pull requests

- https://github.com/elisa-tech/meta-sgl

SGL Software Constellations

- Akin to flavors, spins, editions, etc.
- Configurations to support different use cases.
  - Flight software frameworks.
    - cFS, Fprime, Space ROS, etc.
  - Distribution configurations.
    - Systemd?
    - Minimal or traditional userspace?
    - glibc or musl?
    - Dynamic or static linking?

cFS-One Constellation

(work in progress)

A middle ground between RTOS and classic Linux

- cFS as init/PID 1.
- Bare bones.
  - Minimal kernel config.
  - Easier certification/verification.
- Minimal footprint.
- Safely extensible.
  - Common Libraries, executables.
  - Container runtimes, Virtual machines?

Resources

- Community Repository: Meeting Minutes, and Schedule
- meta-sgl: Linux Distro repository
- Website: Landing Page with all the info on the SIG including the mailing list
- SGL Workshop Videos: YouTube Playlist with all the videos from the first Workshop at Goddard Space Flight Center
- Aerospace WG website (mailing list / meetings)

ELISA
Enabling Linux in Safety Applications

Questions and Comments

Thank you for attending!