



# Open Source Real-Time (for Linux)

Jan Kiszka | Open Source @ 西门子 , November 29, 2023

# Agenda

**Introduction real-time Linux**

**The Xenomai project**

**Siemens' involvement**

**Xenomai versions and technical structure**

**Community aspects**

**Summary**

# Hard Real-Time: Better NOT Miss the Nail

## **Not as fast as possible, as fast as needed**

- Hard real-time not defined by length of reaction time, rather by ensured reaction time
- High damages possible if deadlines are missed

## **Very high reliability of temporal behavior**

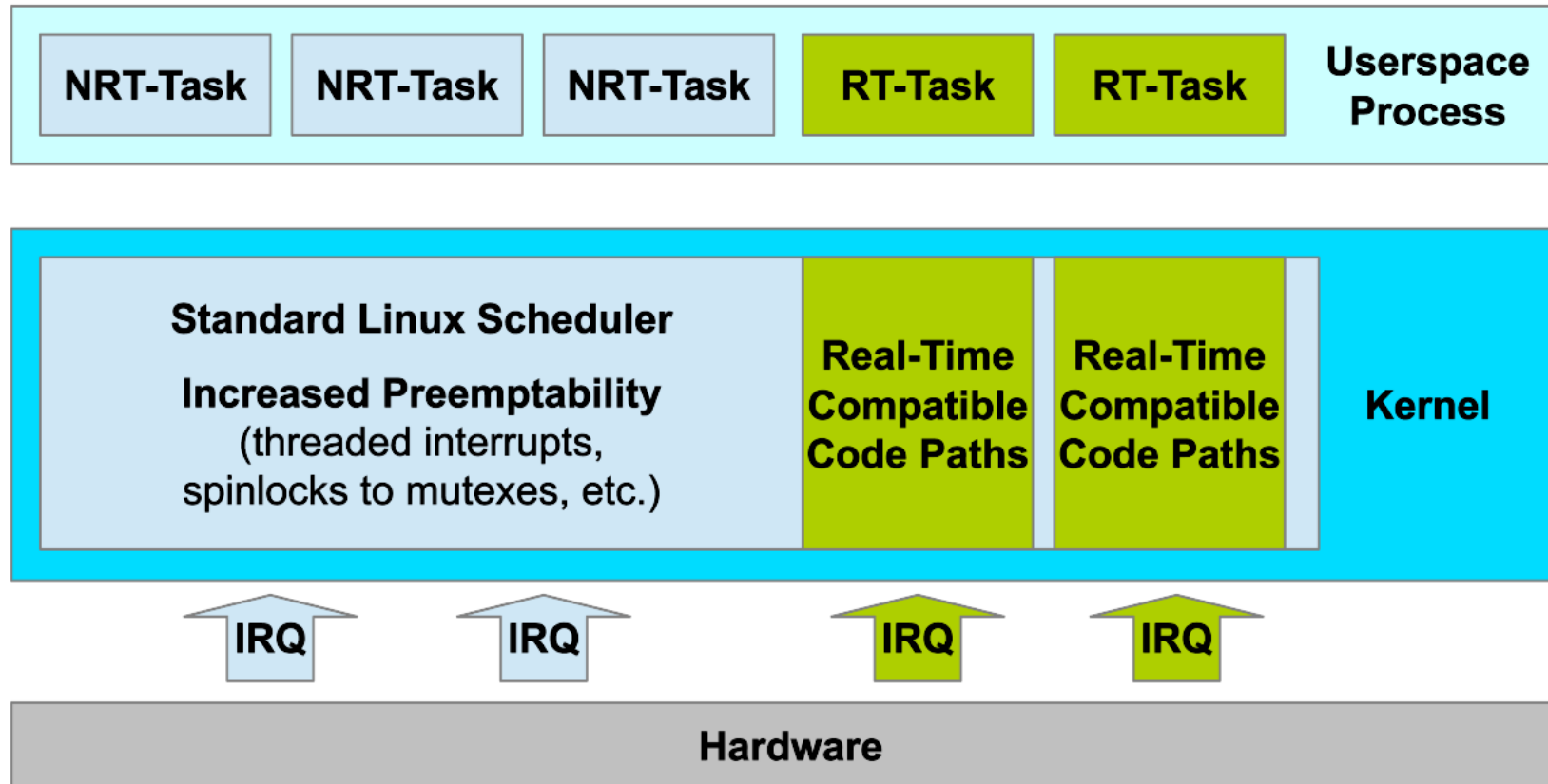
- Aim is avoiding systematic unbounded delays
- Separation of critical parts from non-critical ones
- Performance and/or efficiency step back

## **But: There is nothing like 100% safety**

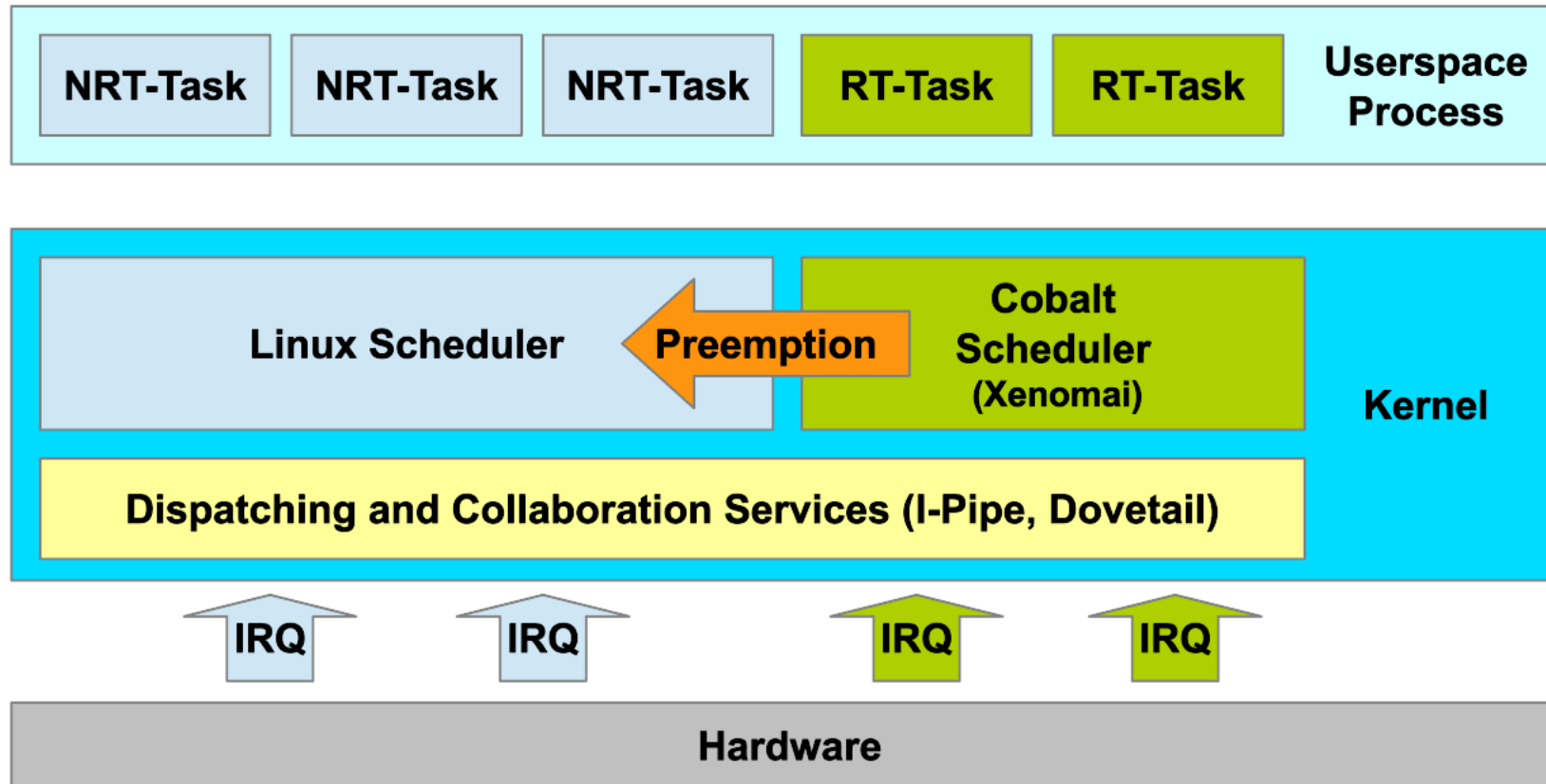
- Many RT applications are large and complex
- So are I/O stacks
- Today's high-performance hardware is non-deterministic

## **So, why not use Linux?**

# Native Real-time for Linux – PREEMPT-RT



# The Co-Kernel Concept of Xenomai



# Why Still Using a Co-Kernel?

## Allows to port complex RT application stacks to Linux

- Different scheduling mechanisms
- Scheduler tunings
- Additional APIs
- Accurate modeling of legacy RTOS behavior

## Architectural separation of RT and non-RT

- Often simpler system configuration
- Application gets early feedback when leaving RT path
- Even libraries cannot disturb RT silently

## Can provide better latencies

- Lower overhead on low-end platforms
- Better latencies with RT and non-RT on same core
- Can decouple from scalability issues (~8+ cores)

# Lifeline of the Xenomai Project

## **Xenomai 1.0**

- Started in 2001 as RTOS portability framework
- Became part of RTAI

## **Xenomai 2.0**

- Stand-alone again since 2005 (different design goals)
- I-pipe patch to hook into the kernel

## **Xenomai 3.0**

- Released in 2015 with renovated co-kernel core („cobalt“)
- Support for RTOS layers over standard Linux / PREEMPT-RT

## **Xenomai 3.2**

- Released in 2021
- New kernel patch (dovetail)

## **Xenomai 4: next generation**



# Xenomai Applications at Siemens



- Hard real-time requirements with minimal latencies
- API-extensions to serve needs of existing (large) code base
- Long-term support



- Maintain separation of RT and non-RT architecturally
- High through-put demands aside RT workload



# Siemens Taking the Lead

**Context: Relatively small contributor community**

**2016: Key contributor passed away**

**2018: Maintainer announced to step back from daily business**

- Wanted to focus on next generation (EVL, Xenomai 4)
- As long-term contributor, Jan Kiszka and, thus, Siemens was asked to take over

**What to do?**

- Migrate products to PREEMPT-RT?
- Take the lead – and invest further?

**Affected business lines decided to enable the latter**

- Jan becomes Xenomai maintainer in late 2018
- 2-4 further Siemens engineers working on upstream topics now (part-time)
- Siemens invests infrastructure and tries to grow the community

**2020: Intel joins as important Xenomai contributor**

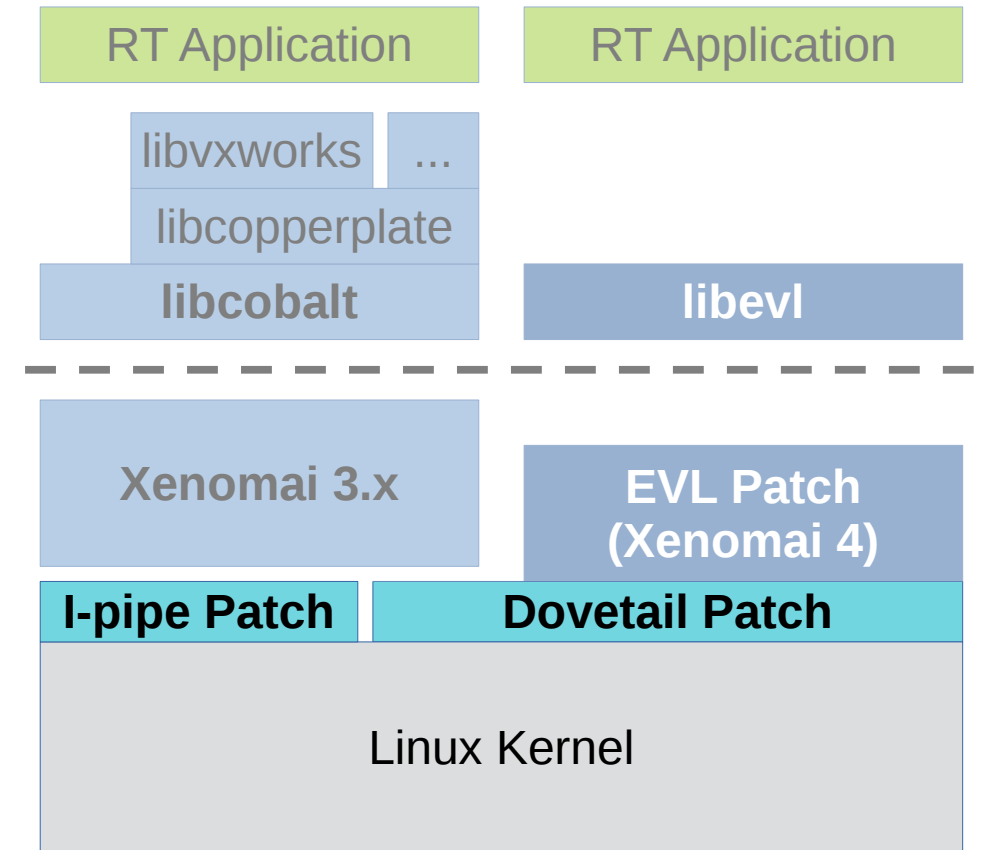
# Foundational Kernel Patches

## Dovetail is the modern patch

- Supports LTS kernels 5.10, 5.15, 6.1, 6.6, and head (currently 6.7-rcX)
- Architectures: ARM, ARM64, x86
- Same for Xenomai 3.2+ and Xenomai 4

## I-pipe is the historic patch

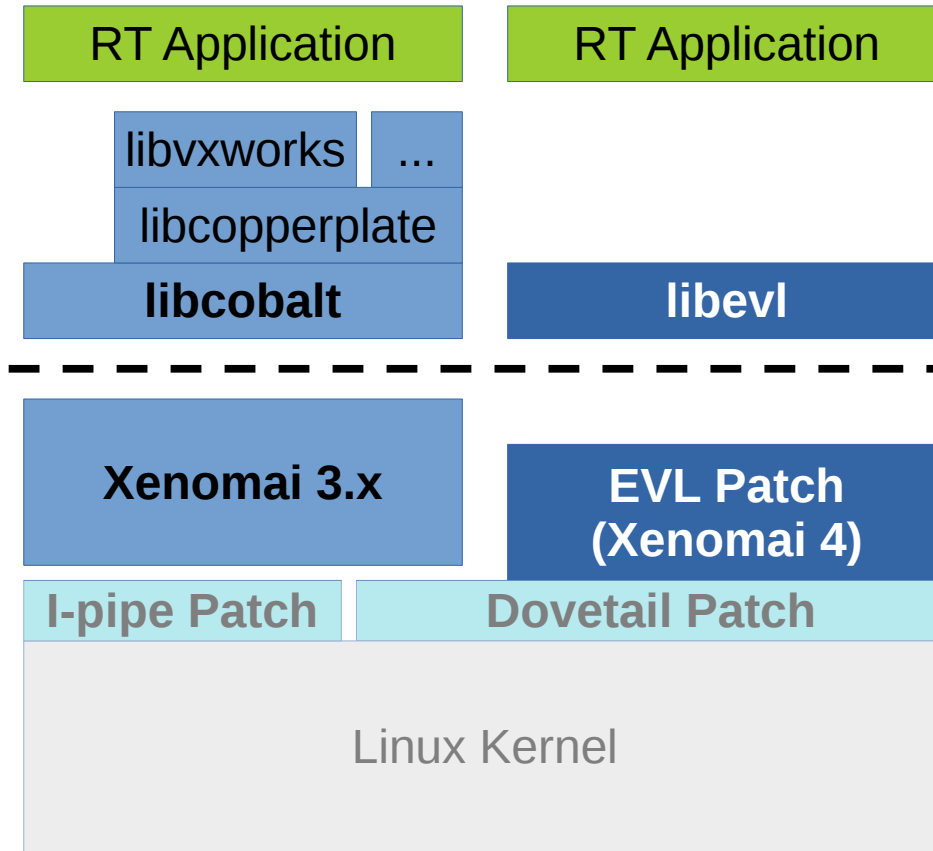
- Ends with 5.4 and Xenomai 3.2.x
- 4.19-cip and 4.4-cip also still maintained



# Xenomai 3 vs. Xenomai 4

## Xenomai 3

- Out-of-source-tree Cobalt core + libraries
- Supports POSIX natively
- Emulates VxWorks, PSOS and “Native” APIs
- Drivers are forks/rewrites against the Real-Time Driver Model (RTDM)
- Currently 3 stable release series
- Maintained by Jan Kiszka



## Xenomai 4

- EVL Core comes as kernel patch
- Supports a new ABI
- No POSIX, no legacy RTOS emulation
- Drivers are kernel drivers with patched-in “Out-of-Band” support
- Aims at supporting 2 LTS kernel series + head
- Maintained by Philippe Gerum

# CI and Testing Infrastructure

## Test matrix is hard to manage

- 3 stable Xenomai releases, 6 kernel series
- 3 architectures
- Real vs. emulated targets
- Debug vs. latency tunings

## How we attack this

- Cornerstone: in-tree smokey test suite and “satellite” tests
- CI “smoke” builds on gitlab.com on Xenomai pushes
- xenomai-images build in AWS cluster
- LAVA test labs at Intel and Siemens

The screenshot displays the LAVA (Linux Automated Validation from Assets) web interface. It is divided into several sections:

- Build:** A list of build targets, each with a green checkmark and a refresh icon. The targets include: arm64-4.19-cip-release, arm64-5.4-release, arm64-5.10-head, arm-4.19-release, arm-5.4-release, arm-5.10-head, i386-userland, powerpc-4.19-release, and x86-4.19-cip-head.
- Test:** A list of test targets, also with green checkmarks and refresh icons. The targets include: trigger-xenomai-images, next, stable/v3.0.x, stable/v3.1.x, and stable/v3.2.x. A dropdown menu is open for 'stable/v3.2.x', showing a 'Child' link and a '#11812' identifier.
- Build Matrix:** A grid of build targets for different architectures and releases, such as build-4.19.beagle-bone-black, build-4.19.hikey, build-4.19.qemu-amd64, build-4.19.qemu-arm64, build-4.19.qemu-armhf, and build-4.19.x86-64-efi.
- Test Matrix:** A grid of test targets corresponding to the build matrix, such as lava-test-4.19.beagle-bone-black, lava-test-4.19.hikey, lava-test-4.19.qemu-amd64, lava-test-4.19.qemu-arm64, lava-test-4.19.qemu-armhf, and lava-test-4.19.x86-64-efi.
- All Devices Table:** A table listing the status of various devices. The table has columns for Hostname, Worker Host, Device Type, state, Health, Is Synced, and Tags.

Hostname	Worker Host	Device Type	state	Health	Is Synced	Tags
beaglebone-black@siemens	lab@siemens	beaglebone-black	Running #3675 xenomai-test-suite [xenomai@siemens]	Unknown	x	
hikey@siemens	lab@siemens	hikey	Running #3679 xenomai-test-suite [xenomai@siemens]	Unknown	x	
qemu01	lab@siemens	qemu	Running #3676 QEMU pipeline, xenomai-images-qemu-armhf [xenomai@siemens]	Good	x	
qemu02	xenomai-lab@intel	qemu	Running #3674 QEMU pipeline, xenomai-images-qemu-armhf [xenomai@siemens]	Good	x	
qemu03	xenomai-lab@intel	qemu	Running #3673 QEMU pipeline, xenomai-images-qemu-armhf [xenomai@siemens]	Good	x	
qemu04	xenomai-lab@intel	qemu	Running #3678 QEMU pipeline, xenomai-images-qemu-arm64 [xenomai@siemens]	Good	x	
x86-64-efi@siemens	lab@siemens	x86	Running #3677 xenomai-test-suite [xenomai@siemens]	Unknown	x	

# The Role of xenomai-images

## Provide quick-start images

- All architectures supported
- EVL support on the way
- QEMU images + generic x86 + BeagleBone Black (ARM) + HiKey (ARM64)
- Debian-based, easy to extend during runtime
- Can be a base for product images (ISAR build tool)
- <https://source.denx.de/xenomai/xenomai-images>

## Provide base images for LAVA tests

- Carry LAVA test descriptions
- Build images (extra packages, kernel + rootfs tarball)
- Describe gitlab CI pipeline (image builds and test runs)



# Building and Strengthening the Community

## Challenges

- Many non-visible users vs. few providing feedback
- Tricky kernel development tasks vs. few key contributors
- Increasing interest in China vs. originally Europe-centric community

## Measures taken so far

- Improve transparency of development workflows
- Try to distribute tasks between more developers
- Bi-weekly community call (open for everyone!)
- Virtual Xenomai meetup in 2022 (>90% from China!)
- Physical meetup in Wuxi on Nov 30, 2023



CC BY-4.0 DEED, Kamin Ginkaew

# Summary

- **The Co-kernel is here to stay**
  - Used in production for >20 years
  - Valid use cases aside PREEMPT-RT
- **Xenomai project is steadily evolving**
  - Xenomai 3: traditional basis for most users
  - Xenomai 4: slimmer, different approach on drivers
- **Take-away: Watch your OSS dependency chain!**
  - Open Source is free to use  
...but should not be free of some own investments
  - You do not need to become a maintainer to support it  
...but be ready if your dependency is not carried by others



Any Questions?

# Thank you!

<http://xenomai.org>

Jan Kiszka <jan.kiszka@siemens.com>  
Xenomai mailing list <xenomai@xenomai.org>