Real-Time OS Overview

Ing. Michal Sojka, Ph.D.

ČVUT v Praze,

michal.sojka (at) cvut.cz

October 25, 2023

Real-Time OS Overview

- There exist hundreds of RT operating systems...
  - more in the past, less today
- VxWorks (WindRiver)
  - Commercial OS
  - Multiprocessor support, optional memory protection, POSIX API, WindAPI, Eclipse-based development environment + tools
  - Flew to Mars
- RTEMS
  - Open-source
  - Multiprocessor support, POSIX API, RTEMS API
  - Active development community
  - Good documentation

Embedded With
RTEMS™
www.rtems.com

Real-Time OS Overview

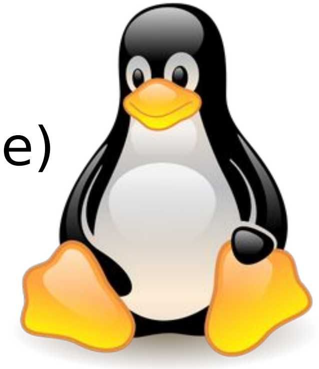
- Zephyr (https://zephyrproject.org/)
  - Modern, open-source RTOS, managed by Linux foundation
    - supported by many hardware vendors
  - Introduced in 2016
  - No memory protection by default (ala VxWorks DKM), user mode with protection (ala VxWorks RTP) available for some hardware.
  - One goal is to provide safety certification as a commercial service

- Apache NuttX
  - Open-source

Real-Time OS Overview

- QNX – commercial, microkernel-based
- Dual-kernel OS (real-time addons to generic OS)
  - RTAI – open-source, Linux runs as a task with lowest priority
  - ~~RT Linux – ditto, patent problem (now Wind River/Intel)~~
  - Ardence/RTX – real-time addon to Windows
- PikeOS (SysGo) – commercial hypervisor, safety critical applications
- FreeRTOS – for small micro-controllers, single address space
- Windows CE – Real-Time OS from Microsoft, Win32 API
- Azure RTOS (ThreadX) – Small RTOS from Microsoft, MS cloud integration
- ~~eCos – professional and open-source version. Development is not too open. Interesting HAL. Offered as an alternative OS for Siemens's PLCs.~~

Linux and real-time



- Standard kernel did not have RT properties
  - Kernels 2.4.x were not preemptive (2.6+ is preemptive)
  - Many companies (MontaVista, TimeSys) tried to turn Linux into RT OS
  - They didn't work with the community – they usually offered old versions
  - Later they hired Ingo Molnar a Thomas Gleixner, to make Linux RT capable and push the patches to the mainline version.
- Around 2004 – birth of PREEMPT_RT patch
  - In the beginning it was distributed as a single huge (unmaintainable) patch to prevent the companies selling untested early development versions.

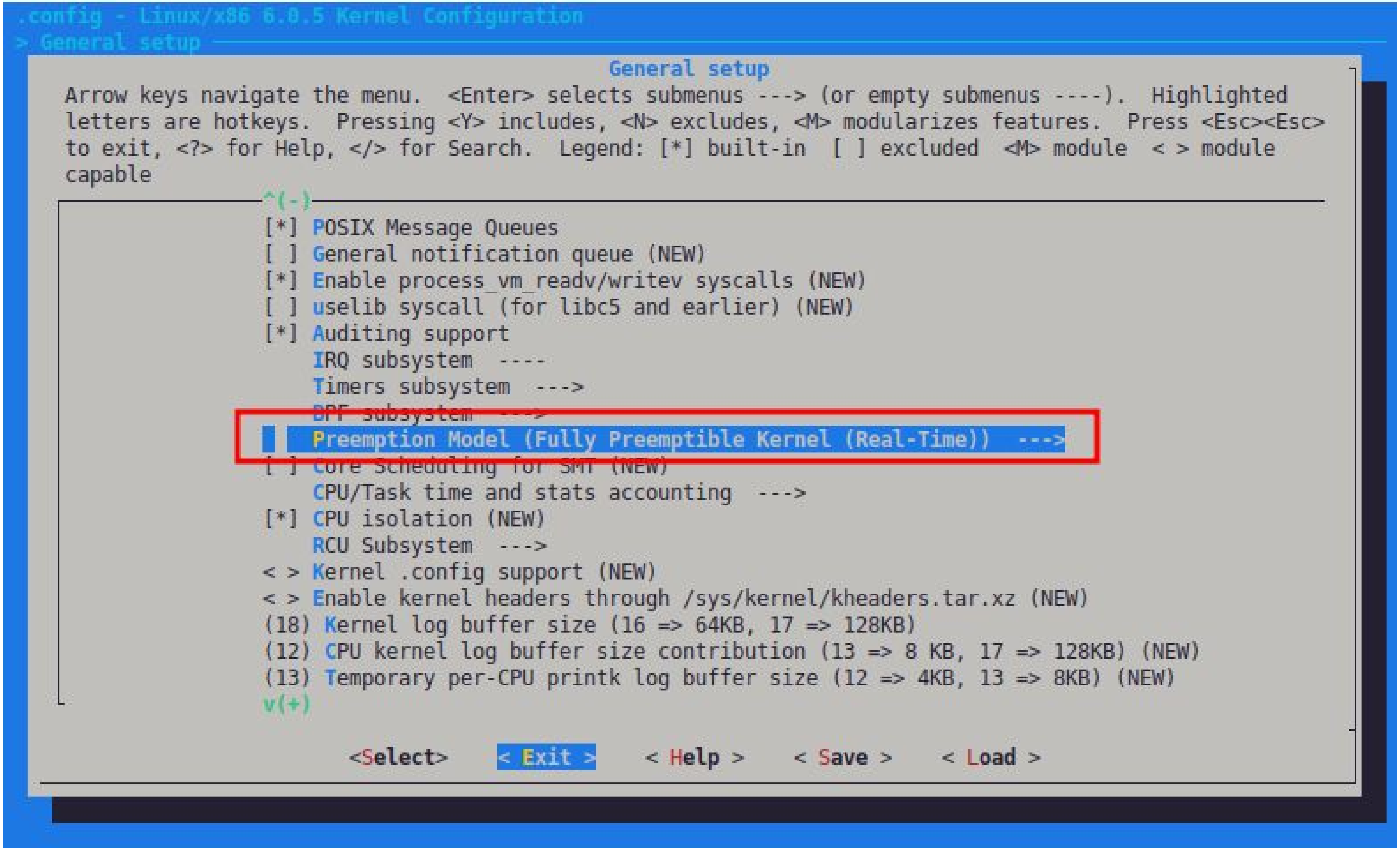
PREEMPT_RT patch

- **Problem**:
  - Even in Linux 2.6.x it was not possible to preempt large amount of kernel code
    - Interrupt handlers, SMP critical sections (spinlock protected), …
    - Those disable interrupts – response time to external events is unnecessary long
- **Solution: rt-preempt patch**
  - Most spinlocks are replaced by mutexes (critical sections become preemptive)
  - Implements priority inheritance (prevents priority inversion)
  - IRQ handlers and softirqs are converted to threads (become preemptive)
  - Timers were reworked to provide high resolution

PREEMPT_RT current status

- Currently, most of the „patch“ is already in the mainline.
  - Linux 5.15 (released last year) includes last major piece (real-time locking – spinlock replacement with RT mutexes)
  - A few pieces (~ 80 changes) is still outside... final merge date is unknown :-)
- In 2015, the preempt-rt project was funded by Google (via Linux Foundation)
- Writing RT applications for Linux is simple:
  - https://wiki.linuxfoundation.org/realtime/documentation/howto/applications/application_base
  - Use mlockall(), SCHED_FIFO scheduler and “page-in” your stack.

Real-Time preempt configuration

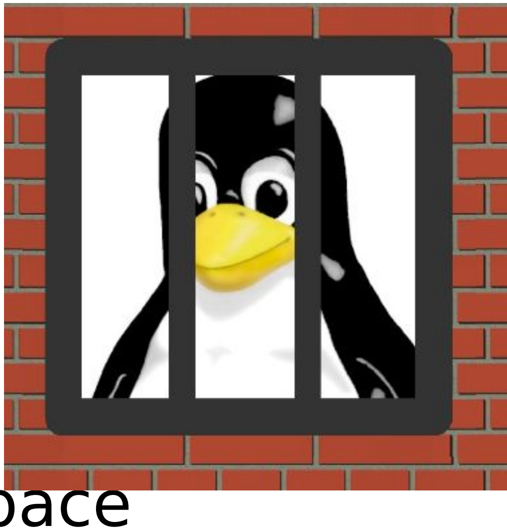


```
.config - Linux/x86 6.0.5 Kernel Configuration
> General setup ─────────────────────────────────────────────────────────────────
                                  General setup
  Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty submenus ----).  Highlighted
  letters are hotkeys.  Pressing <Y> includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc>
  to exit, <?> for Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < > module
  capable
                ^(-)
                [*] POSIX Message Queues
                [ ] General notification queue (NEW)
                [*] Enable process_vm_readv/writev syscalls (NEW)
                [ ] uselib syscall (for libc5 and earlier) (NEW)
                [*] Auditing support
                    IRQ subsystem  ----
                    Timers subsystem  --->
                    BPF subsystem  --->
                    Preemption Model (Fully Preemptible Kernel (Real-Time))  --->
                [ ] Core Scheduling for SMT (NEW)
                    CPU/Task time and stats accounting  --->
                [*] CPU isolation (NEW)
                    RCU Subsystem  --->
                < > Kernel .config support (NEW)
                < > Enable kernel headers through /sys/kernel/kheaders.tar.xz (NEW)
                (18) Kernel log buffer size (16 => 64KB, 17 => 128KB)
                (12) CPU kernel log buffer size contribution (13 => 8 KB, 17 => 128KB) (NEW)
                (13) Temporary per-CPU printk log buffer size (12 => 4KB, 13 => 8KB) (NEW)
                v(+)

                 <Select>    < Exit >    < Help >    < Save >    < Load >
```

Other Linux-related solutions

- **SCHED_DEADLINE** – Linux EDF scheduler (in mainline)
  - Implements Constant Bandwidth Server (CBS) that provides temporal task isolation (protects WCET from overruns)
  - https://github.com/jlelli/sched-deadline
  - http://www.evidence.eu.com/sched_deadline.html



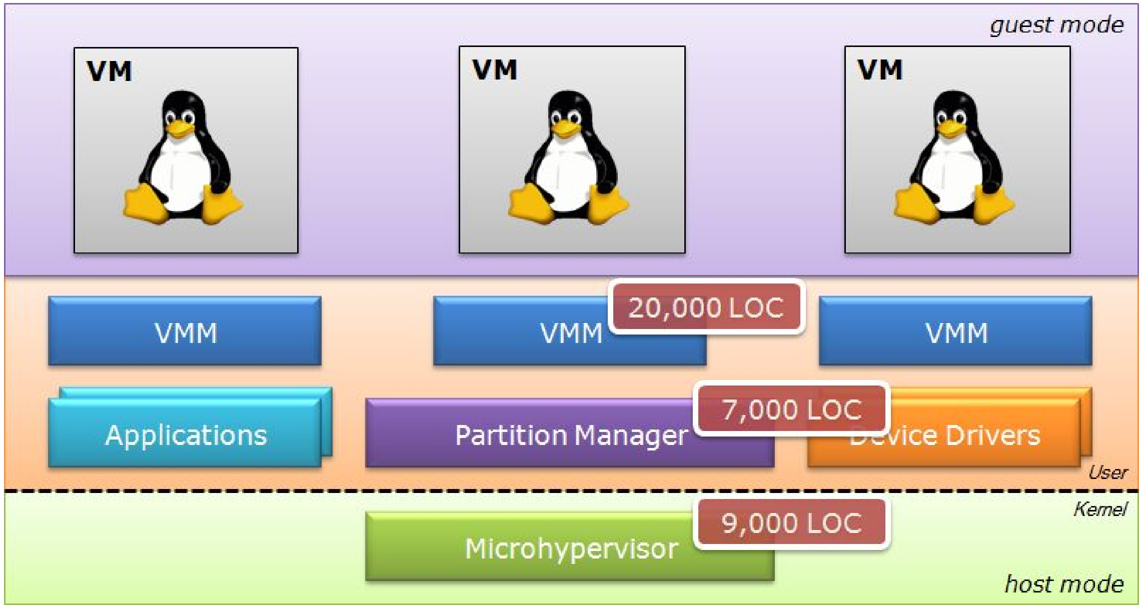
- **JailHouse** – partitioning hypervisor for Linux
  - Small hypervisor for real-time safety critical applications
  - https://github.com/siemens/jailhouse
- **Xenomai** – hard real-time from Linux user space
  - Adeos – IRQ and syscall virtualization

Linux rt-preempt – links

- Real-Time Linux Wiki:
  https://wiki.linuxfoundation.org/realtime/start
  (older site: http://rt.wiki.kernel.org)
- Mailing list: linux-rt-users@vger.kernel.org
- OSADL: http://www.osadl.org
- „Latest stable“ Real-Time Linux
  http://www.osadl.org/Latest-Stable.latest-stable-realtime-linux.0.html
- Linux Weekly Newsletter: http://lwn.net/

Microkernels (TU Dresden)

- L4Re (https://l4re.org/)
  - L4 microkernel + run-time environment
  - Currently used in VW cars etc.
- NOVA Microhypervisor (http://hypervisor.org)
  - Small Trusted Computing Base
- Genode OS framework
  - Unified user space for microkernels
  - Cool!
  - UNIX emulation…


guest mode
VM
VM
VM
VMM
VMM
20,000 LOC
VMM
Applications
Partition Manager
7,000 LOC
Device Drivers
User
Kernel
Microhypervisor
9,000 LOC
host mode