Porting RTAI over Adeos

http://www.aero.polimi.it/~rtai/

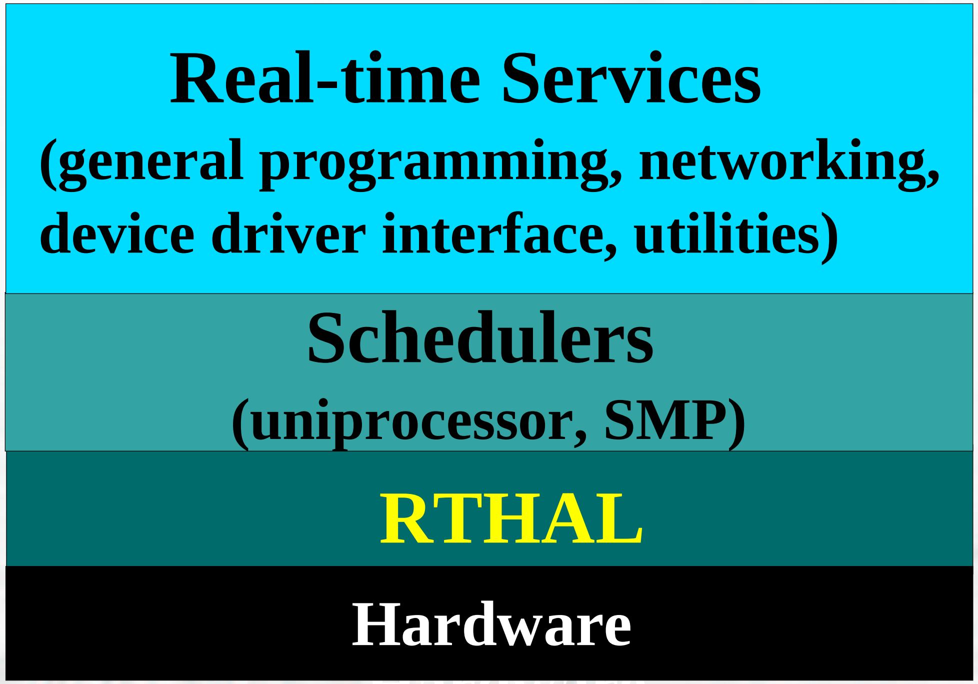
http://freesoftware.fsf.org/projects/adeos/

FOSDEM

Brussels, February 2003

Philippe Gerum, rpm@xenomai.org

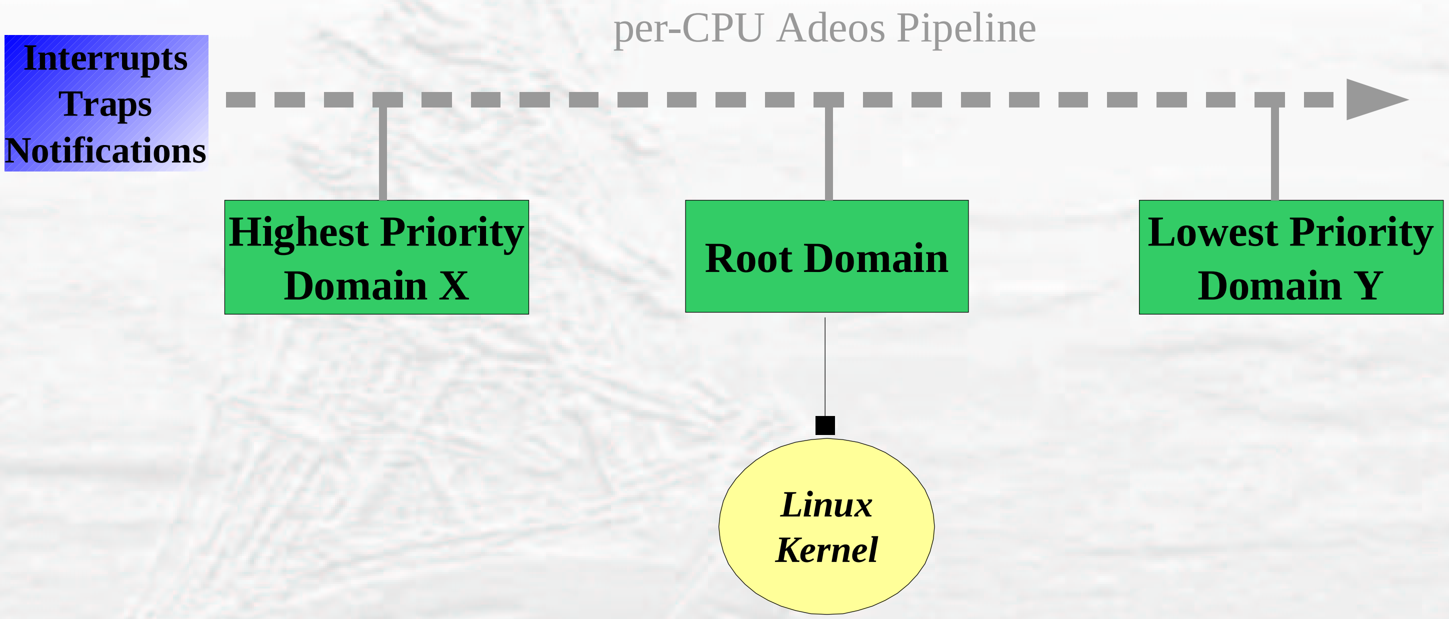
RTAI architecture


Real-time Services
(general programming, networking, device driver interface, utilities)
Schedulers
(uniprocessor, SMP)
RTHAL
Hardware

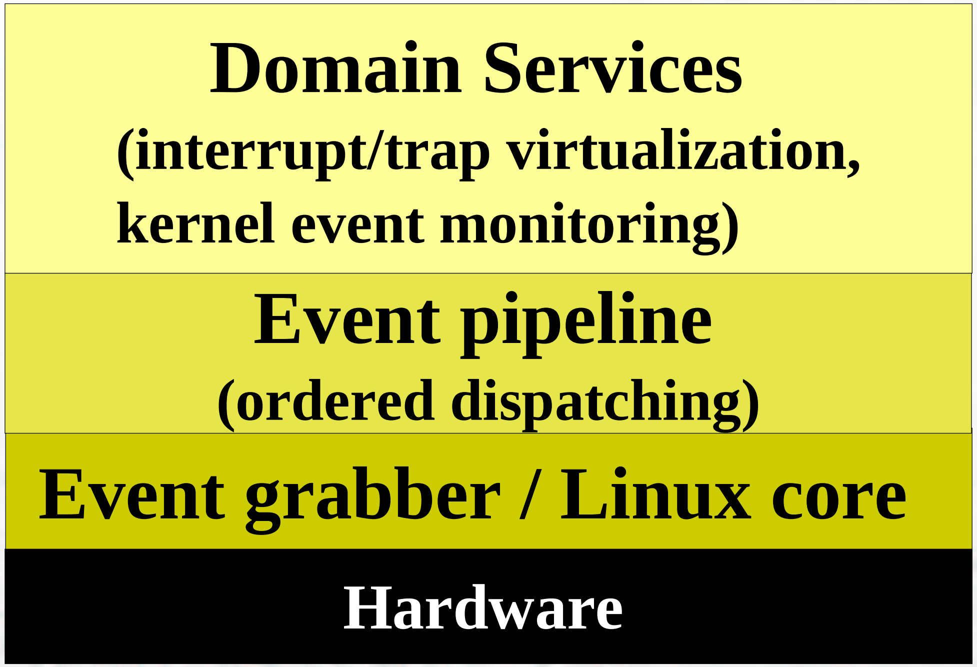
Hardware

- ➢Real-time APIs
- ➢Real-time tasks scheduling
- ➢Hardware control (interrupt management)

Adeos nanokernel scheme

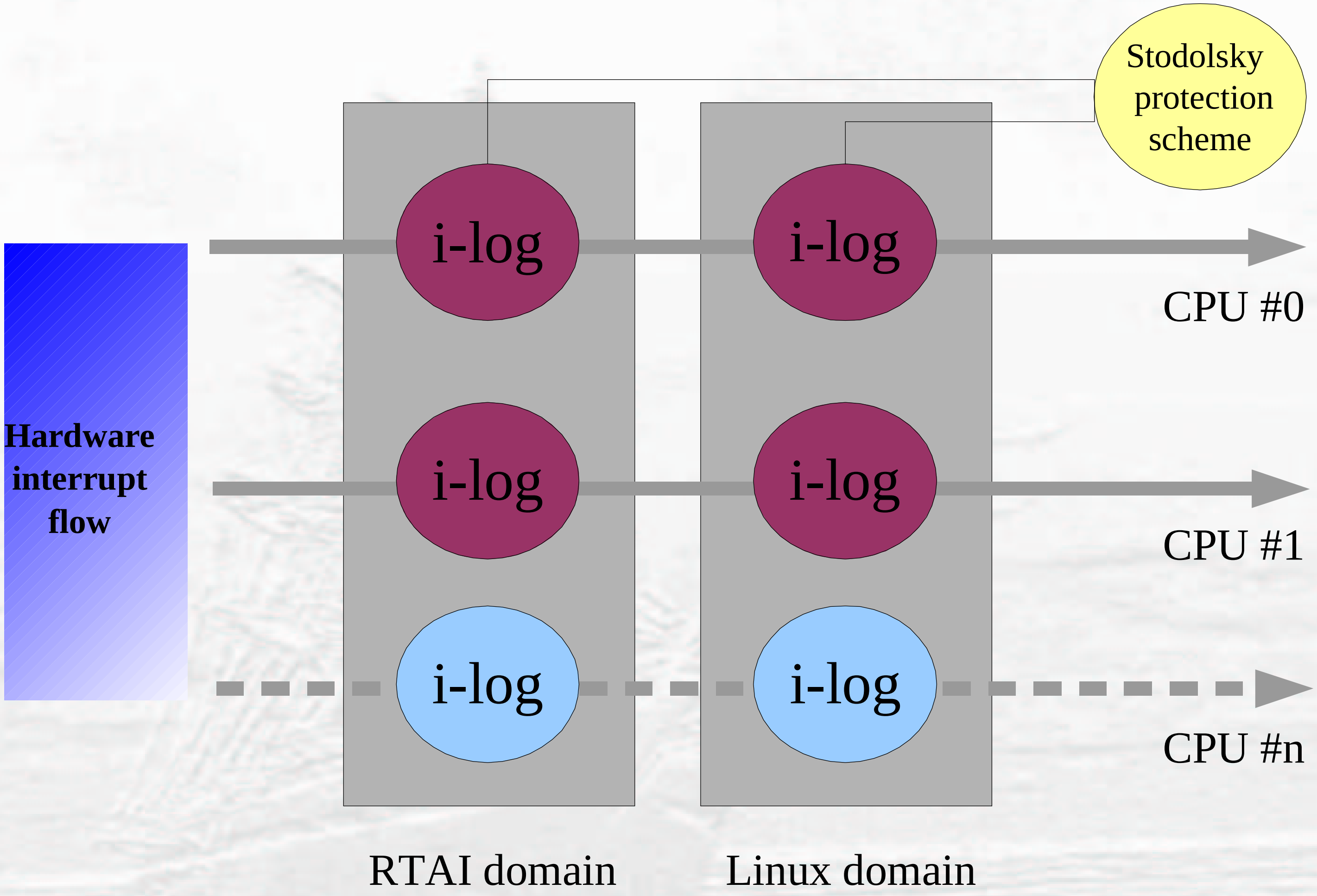


Adeos internals



- ➢Domain programming API
- ➢Event ordering and dispatching
- ➢Low-level interposition

Optimistic interrupt protection



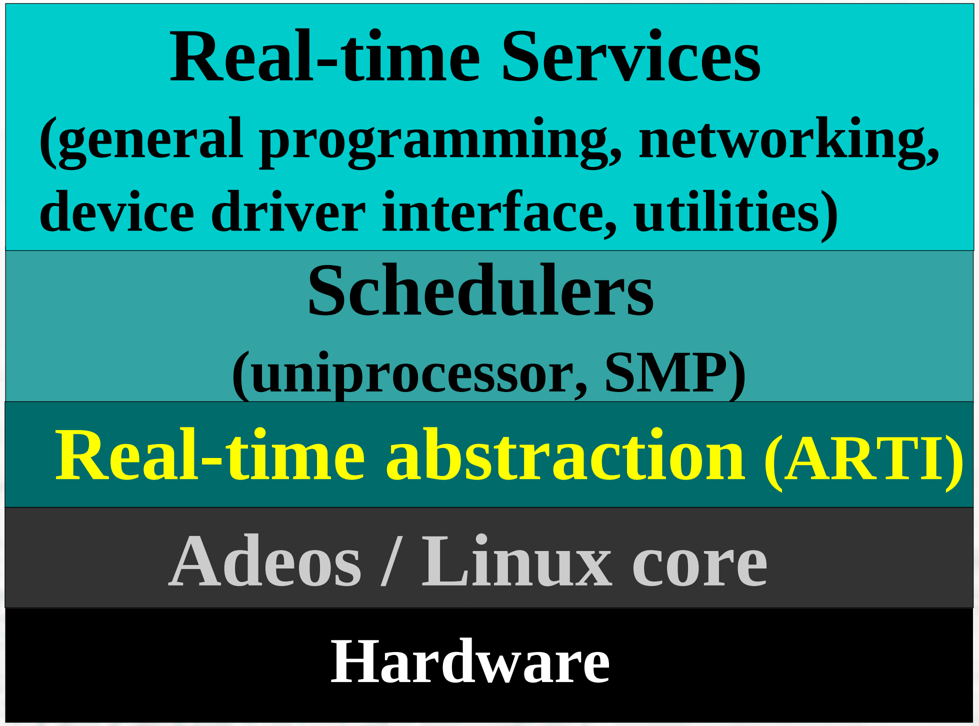
Integration goals

- RTHAL / RT-services decoupling
  - ✓ Maintenance
  - ✓ Linux 2.6
- Immediate RTAI enhancements
  - ✓ APIC/IO-APIC support for uniprocessor
  - ✓ Extended LXRT portability
  - ✓ No conflict with profiling code
- Demanding Adeos use case

Integration constraints

- Real-time determinism
- No functional change in RTAI
- Complete Kernel and User space support
- No API change in RTAI
- Adeos/RTHAL interchangeability
- Adeos genericity

RTAI domain over Adeos



Real-time Services

(general programming, networking, device driver interface, utilities)

Schedulers

(uniprocessor, SMP)

Real-time abstraction (ARTI)

Adeos / Linux core

Hardware

- Real-time APIs
- Real-time tasks scheduling
- Core real-time services

RTHAL services

- Real-time interrupt management
  - ✓ Virtualization (software PIC)
  - ✓ SMP affinity
  - ✓ Propagation
- Timer control
  - ✓ x86 APIC (SMP only)
  - ✓ x86 8254 PIT
- Linux service request system

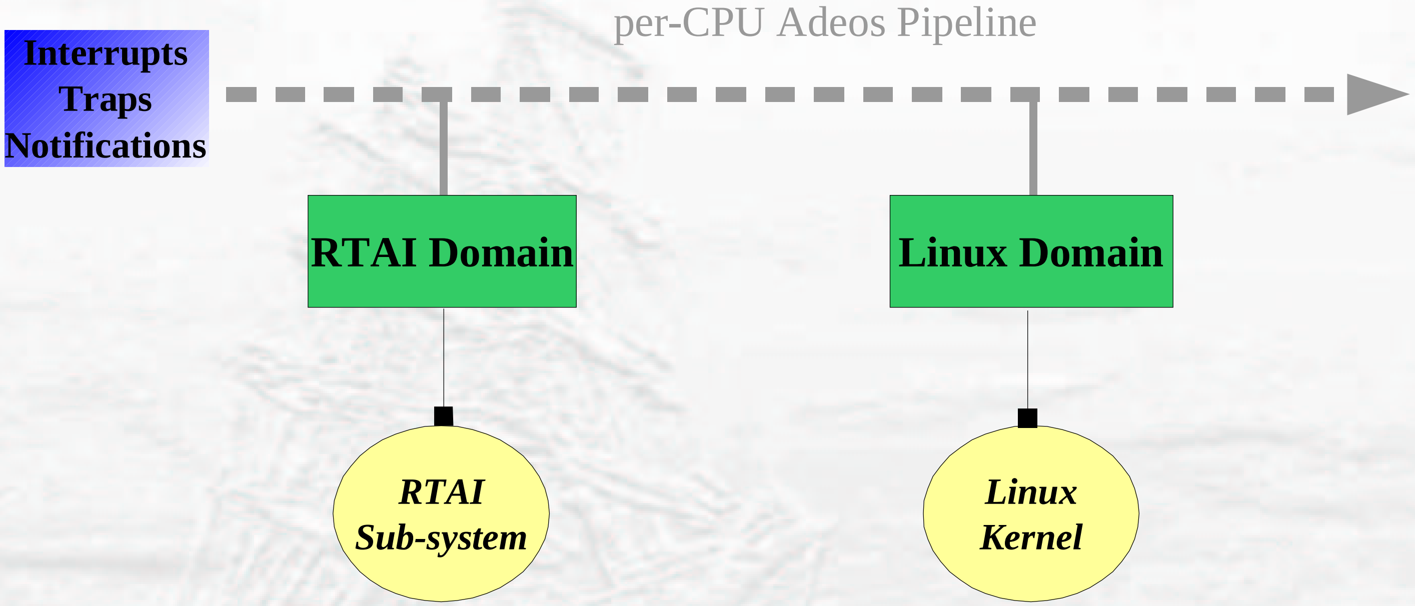
Adeos services

- Per-domain event virtualization
  - ✓ Hardware interrupts
  - ✓ Processor traps
  - ✓ Linux kernel notifications
- Pipeline control
  - ✓ Domain registration
  - ✓ Event propagation (accept, pass, terminate, discard)
  - ✓ Stage stalling/unstalling (Stodolsky-based)
- Inter-domain mutexes (w/ priority inheritance)

Adeos-based Real-Time Interface

- Real-time interrupt management
  - ✓ Adeos pipeline services
- Timer control
  - ✓ x86 APIC (UP and SMP)
  - ✓ x86 8254 PIT
- Linux service request system
  - ✓ Adeos virtual interrupts

Adeos-based RTAI



Adeos impact on RTAI

- RTHAL/ARTI performance comparison

(overloaded Dual PIII 750 Mhz)	*Max. Latency calibration (kernel)*	*Max. Latency calibration (user)*	*Max. Stress (kernel)*	*Max. Task switches (kernel)*
RTHAL	**14 µs**	**30 µs**	**27 µs**	**1.2 µs**
ARTI	**20 µs**	**42 µs**	**27 µs**	**1.3 µs**

- Changes in RTAI code
  - RTHAL fully rewritten ☚ ARTI over Adeos
  - 34 / 750 source files changed
  - < 100 lines affected / 150,000 in the APIs
  - No API change

Current status

- Adeos-enabled RTAI 24.1.11-pre1 released
- Iso-functional (kernel / user space)
- x86 port only
- Uniprocessor and SMP

Possible extensions

- User space real-time enhancements (LXRT)
  - ✓ Improved portability
  - ✓ Smoother syscall integration
  - ✓ Simpler hard/soft real-time transitions
- Debugging and instrumentation
  - ✓ Patchless debugger
  - ✓ Event logging facility
  - ✓ Perturbation generator
- Concurrent RTOS in a single Linux system