

**PPGEAS - Tempo real II**

**RT OS**

**and**

**RT Linux**

Daniel Bristot de Oliveira

Who I'm and what we will  
talk about?

Operating System overview;  
RT Operating Systems; and  
RT Linux

# Operating System overview

(for non OS people)



Application

The diagram consists of three stacked rectangular layers. The top layer is yellow and contains the text 'Application'. The middle layer is orange and contains the text 'Operating System'. The bottom layer is light blue and contains the text 'Hardware'. Each layer is separated by a thin black border.

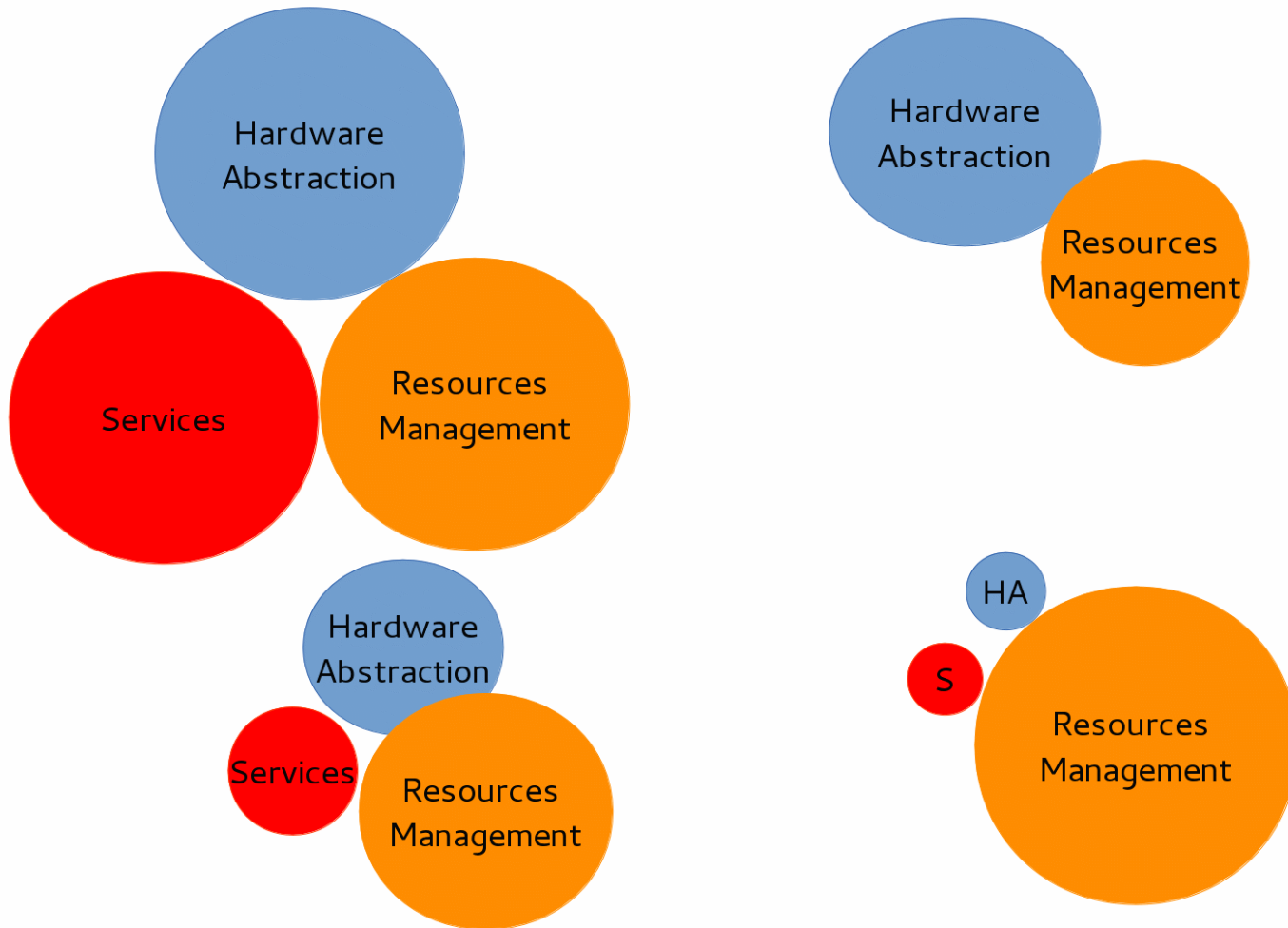
Operating System

Hardware

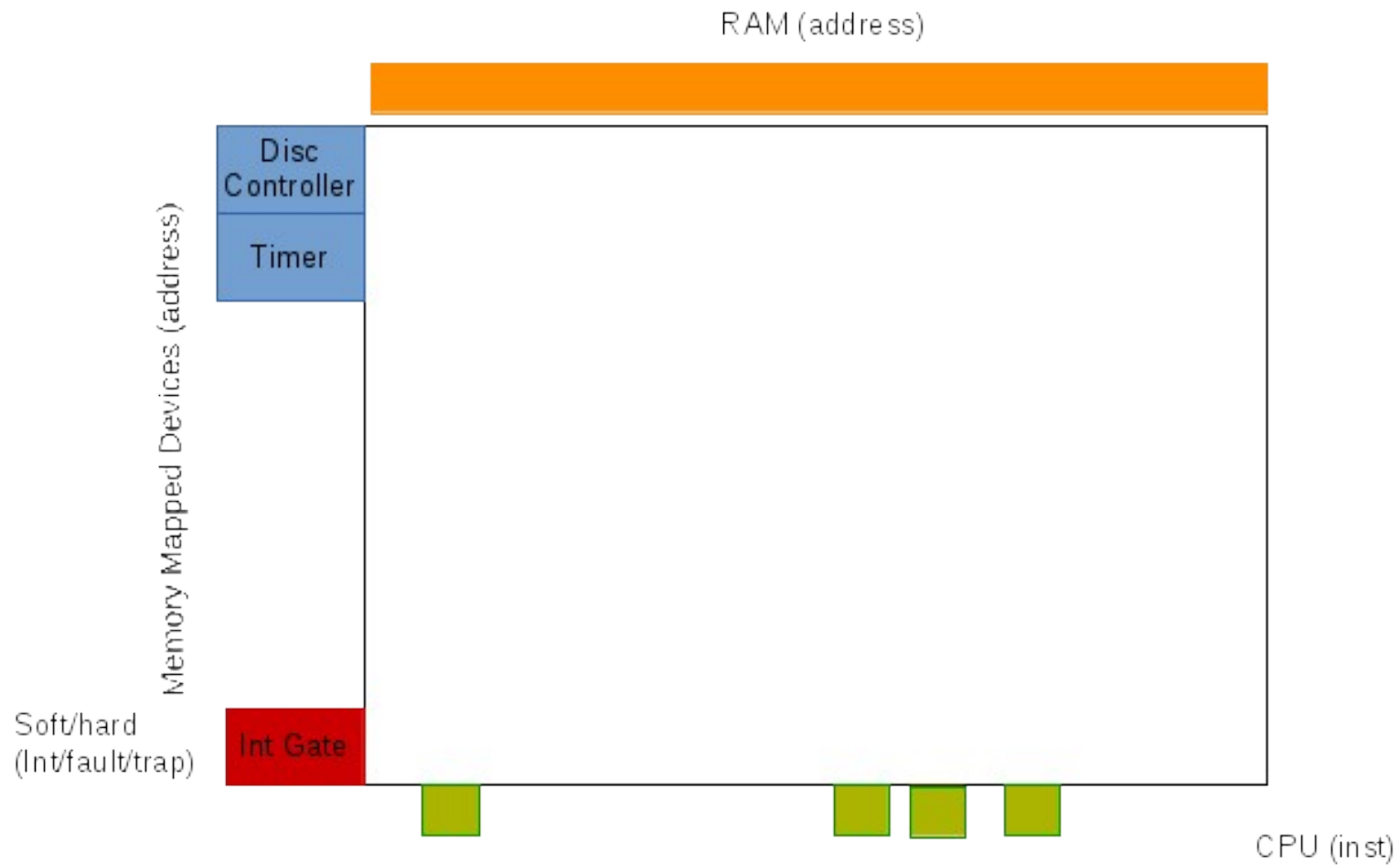
# Redefining the Operating System

## overview

# Tasks of an OS

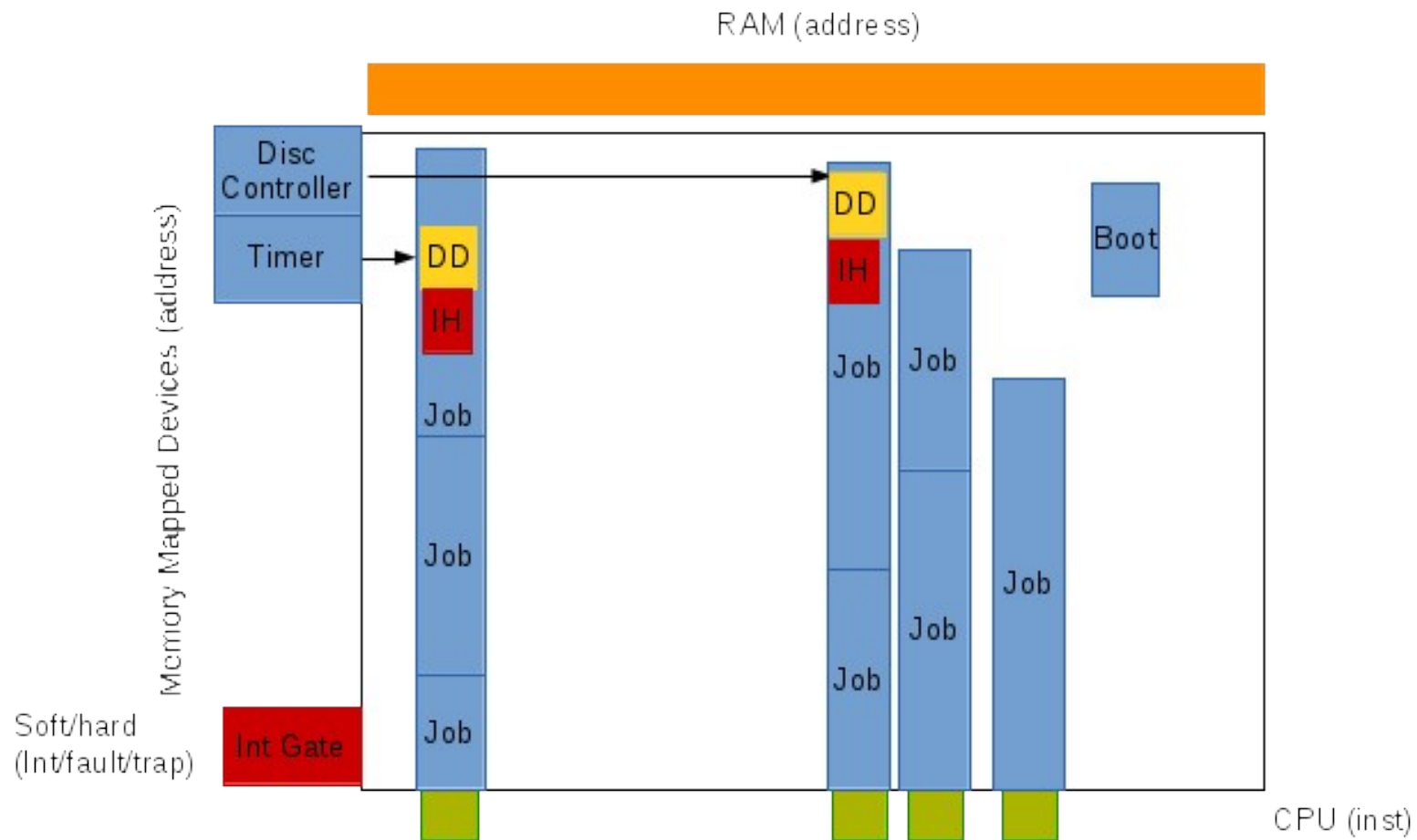


# Hardware...





# Embedded system without OS



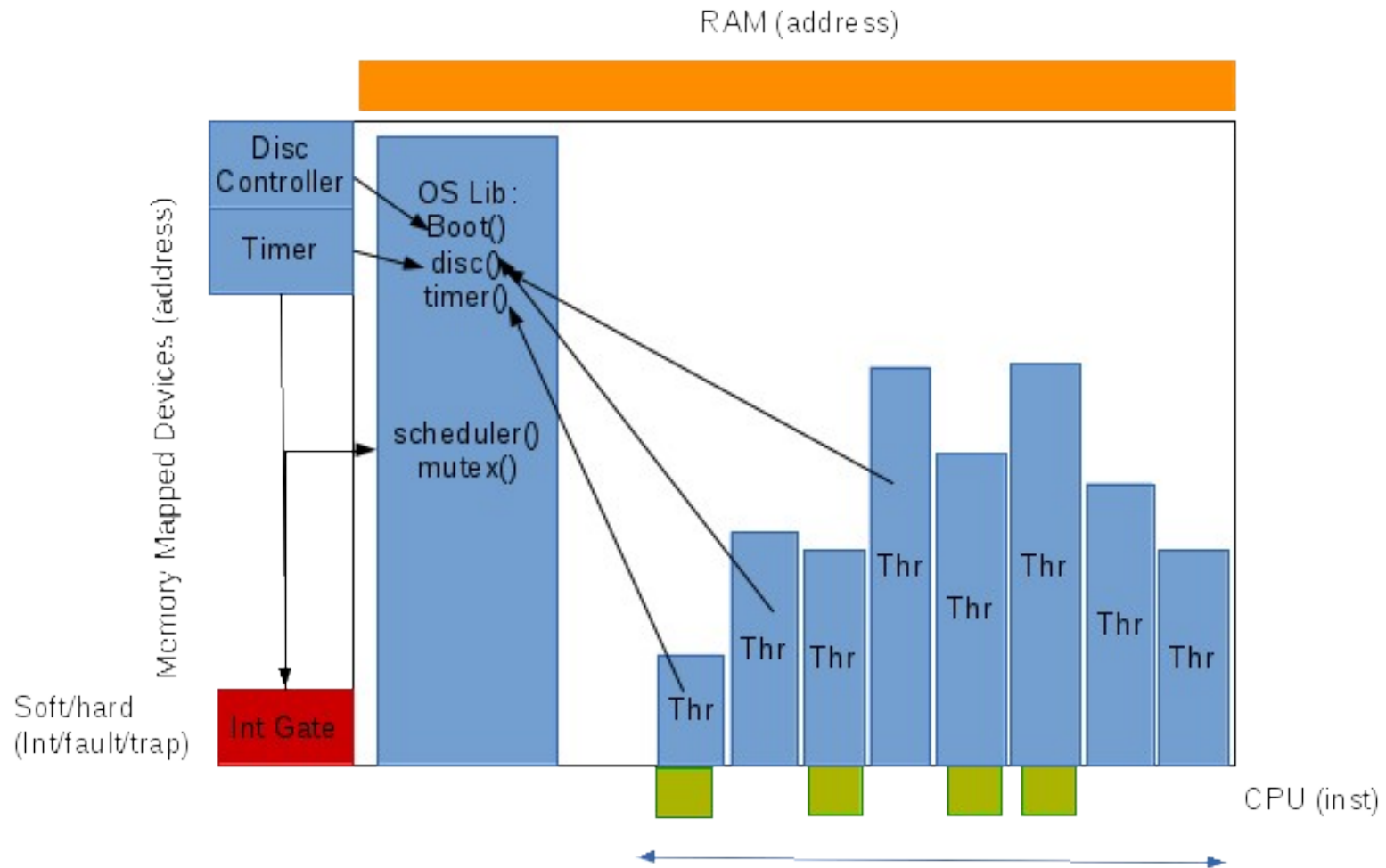
# OS as a library

(hardware abstraction)



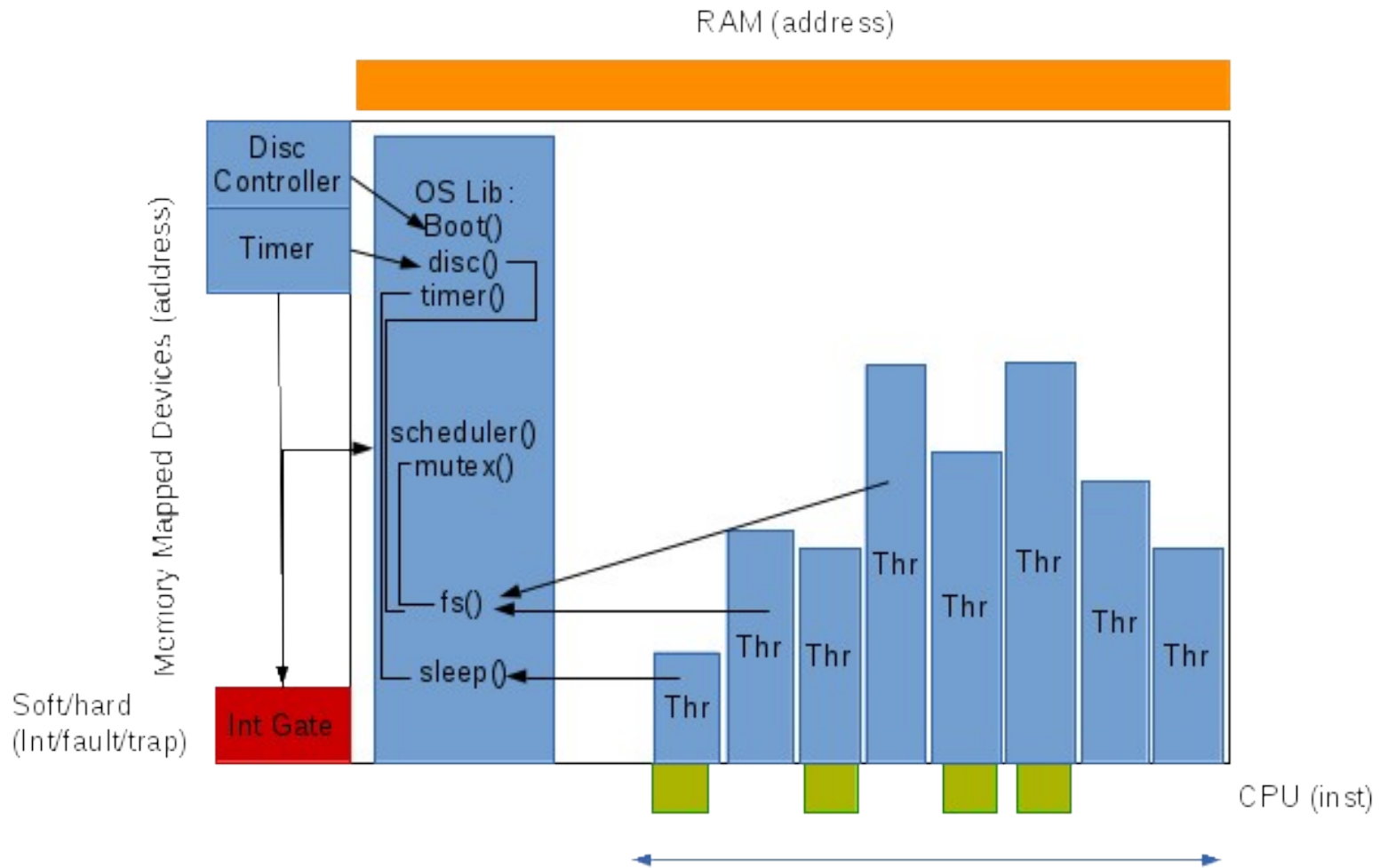
# OS as a library

(+ resource management)

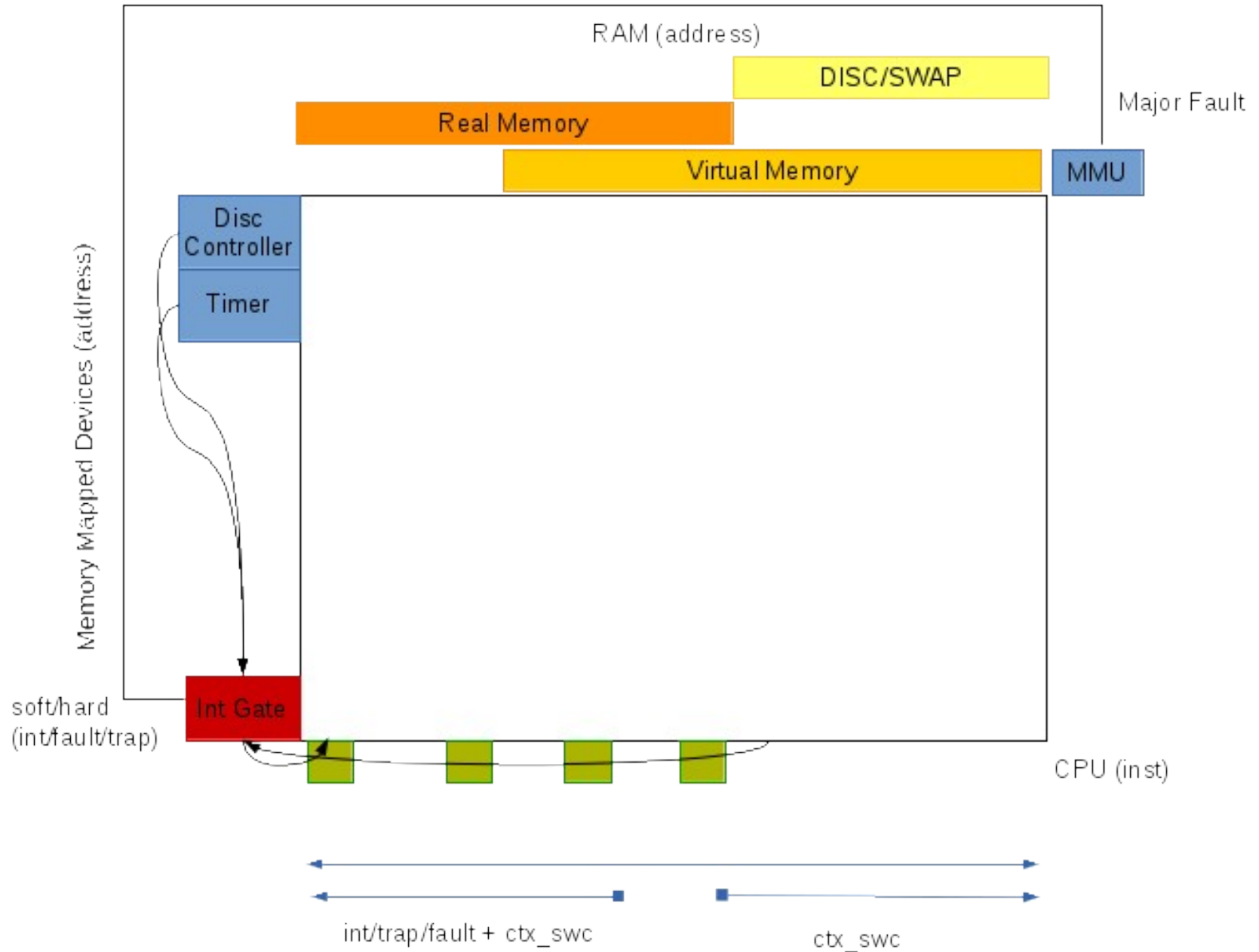


# OS as a library

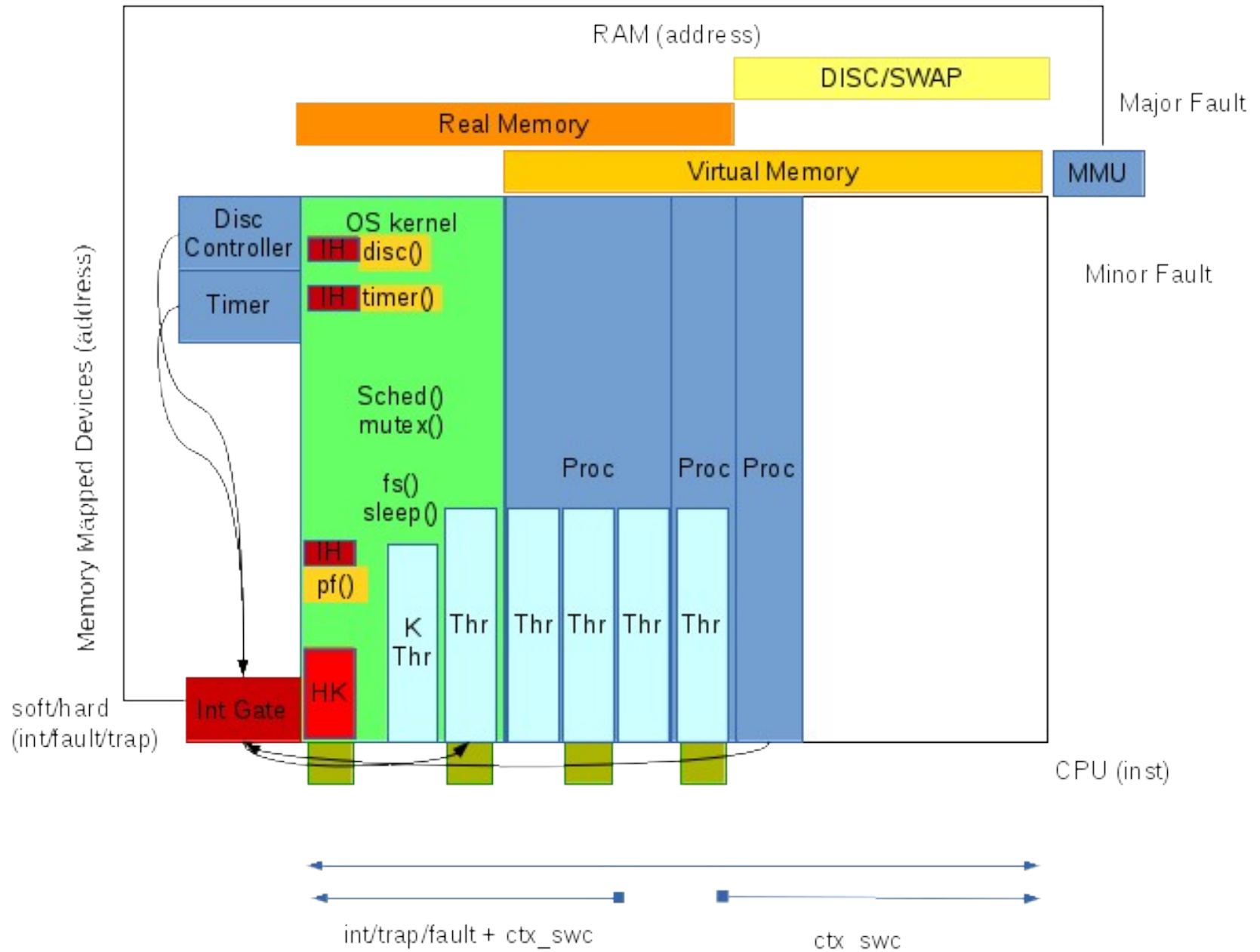
(+ services)



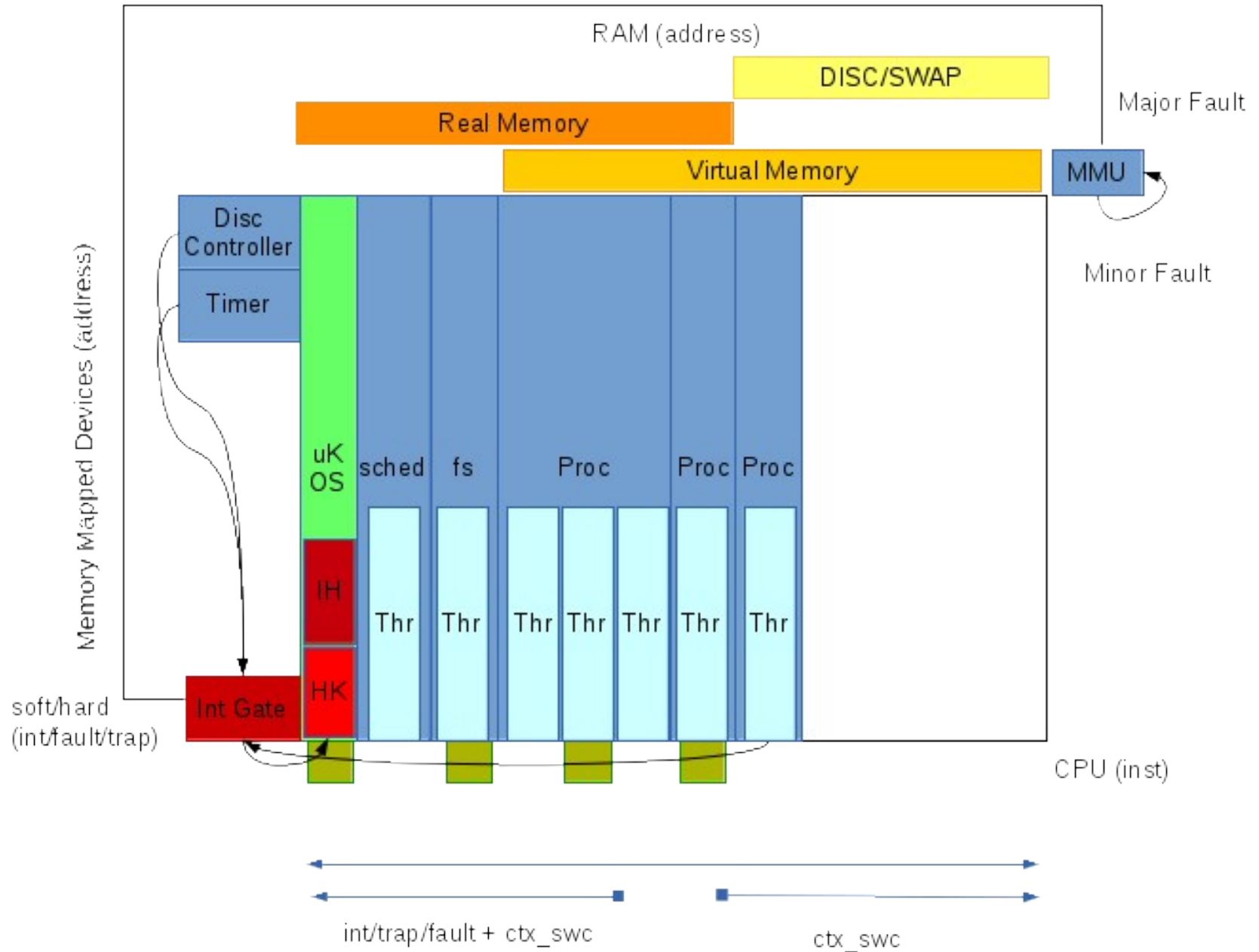
# Memory Protection - MMU



# OS with Kernel



# OS with Micro-Kernel



A context switch: jumping to RT

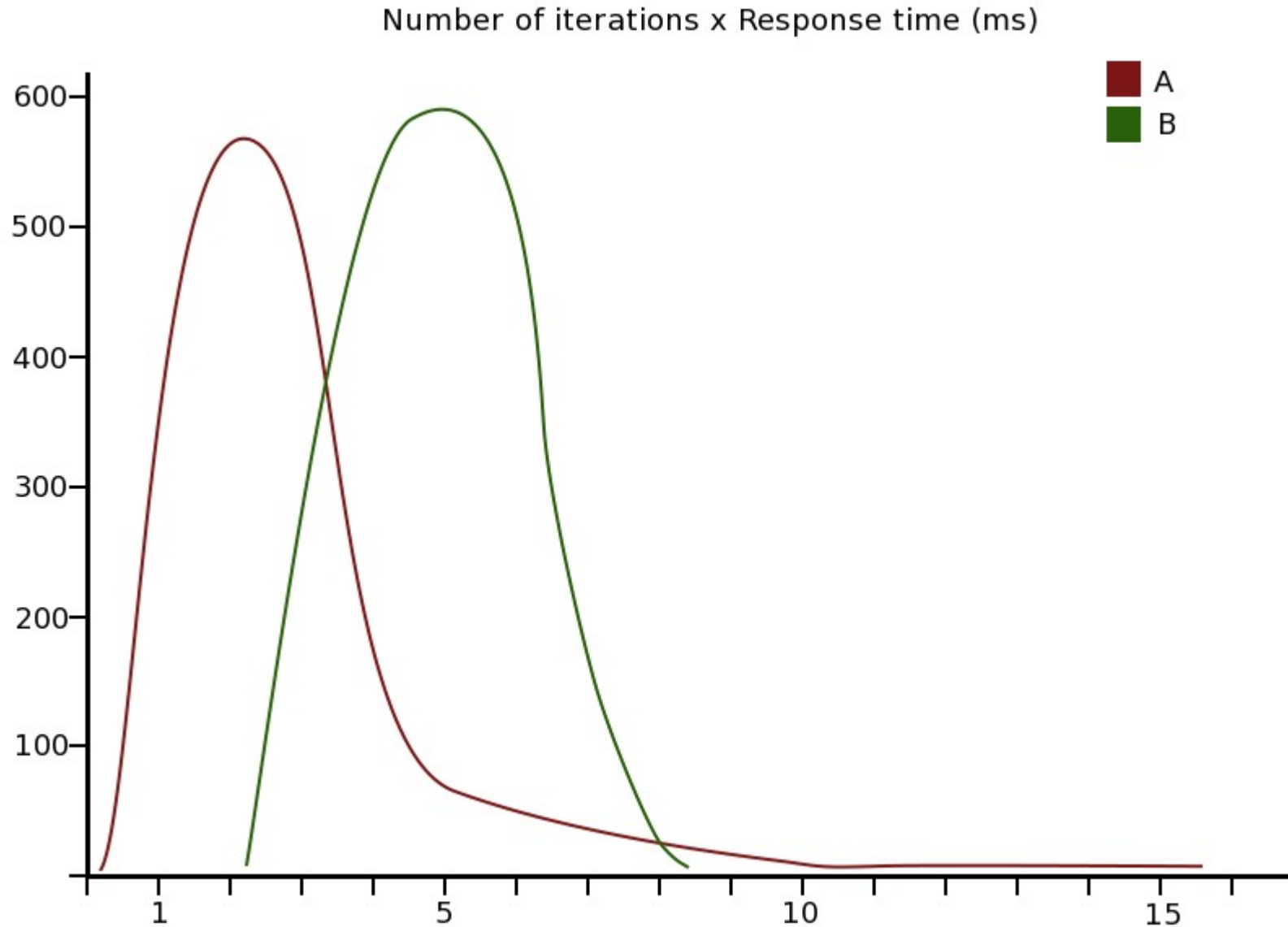


What does Real-Time  
means?

Event -> Response,  
latency,  
response time,  
deadline...

Real-Time means  
fast?

# Real Time != FAST!



RT means

Determinism,

WCE,

Response Time.

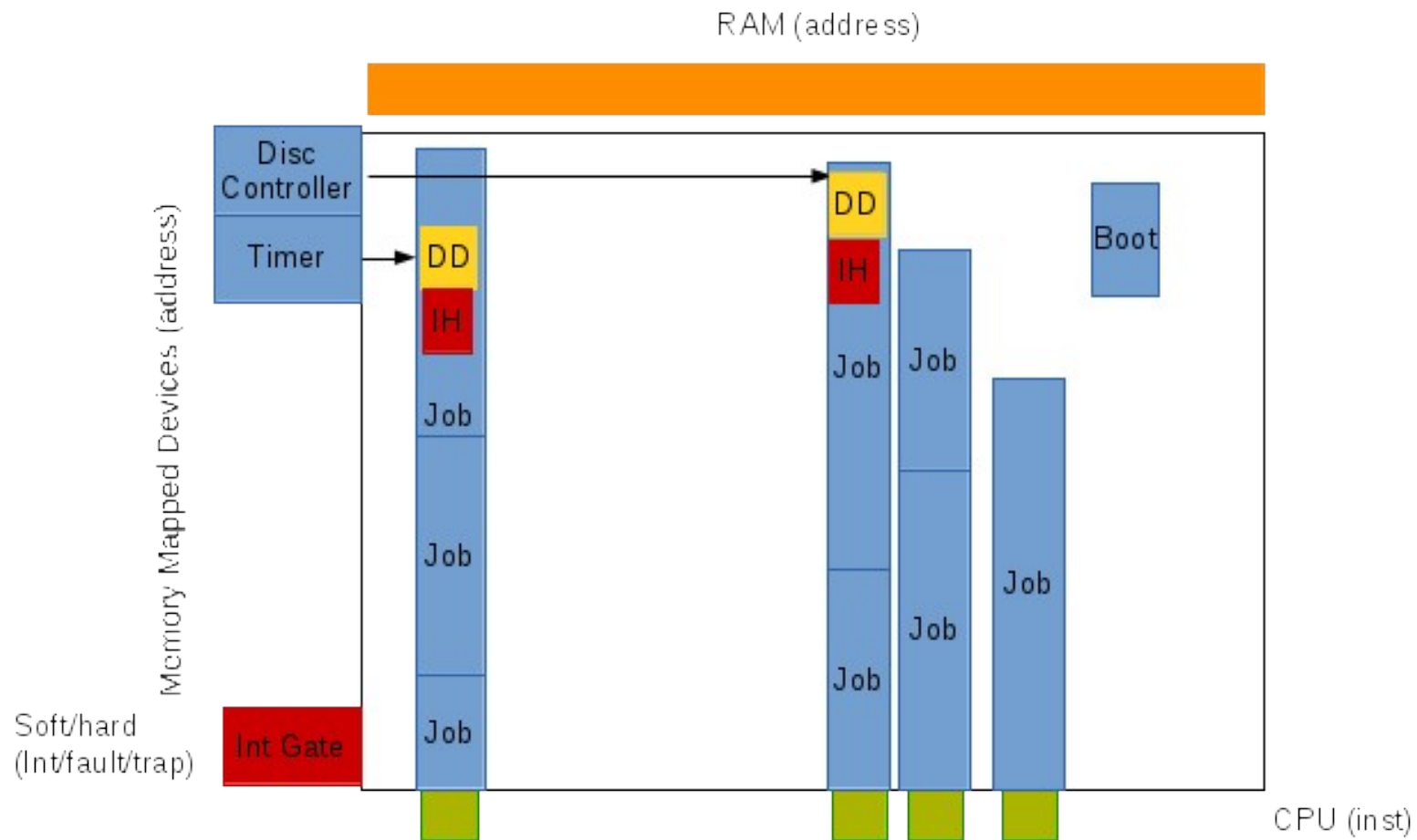
What is a Real-Time  
Operating System?

Deterministic hardware  
+  
Deterministic hardware abstraction  
+  
Services with WCET  
+  
Deterministic Resource Management

# Real-Time Operating System Examples



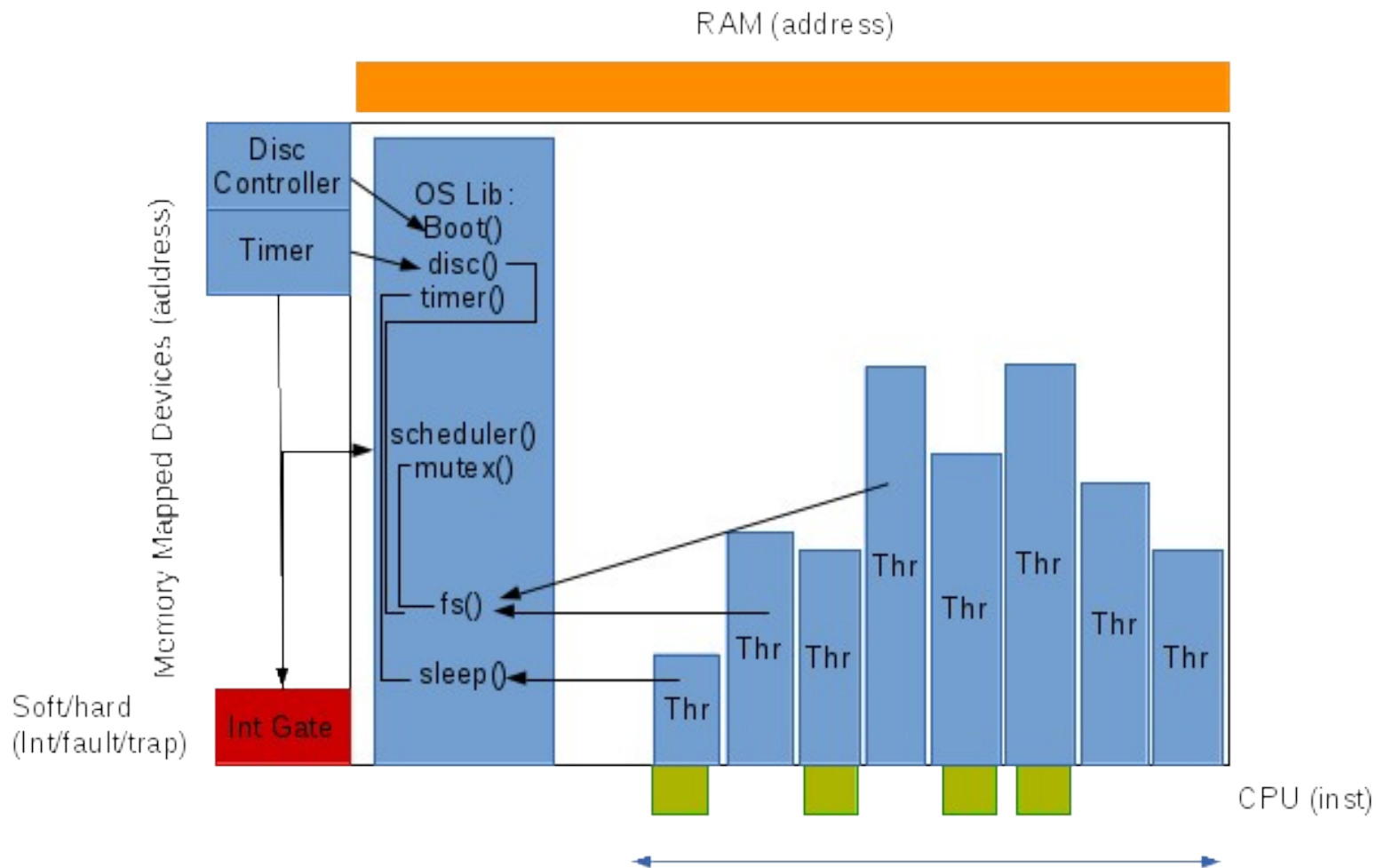
# Embedded system without OS



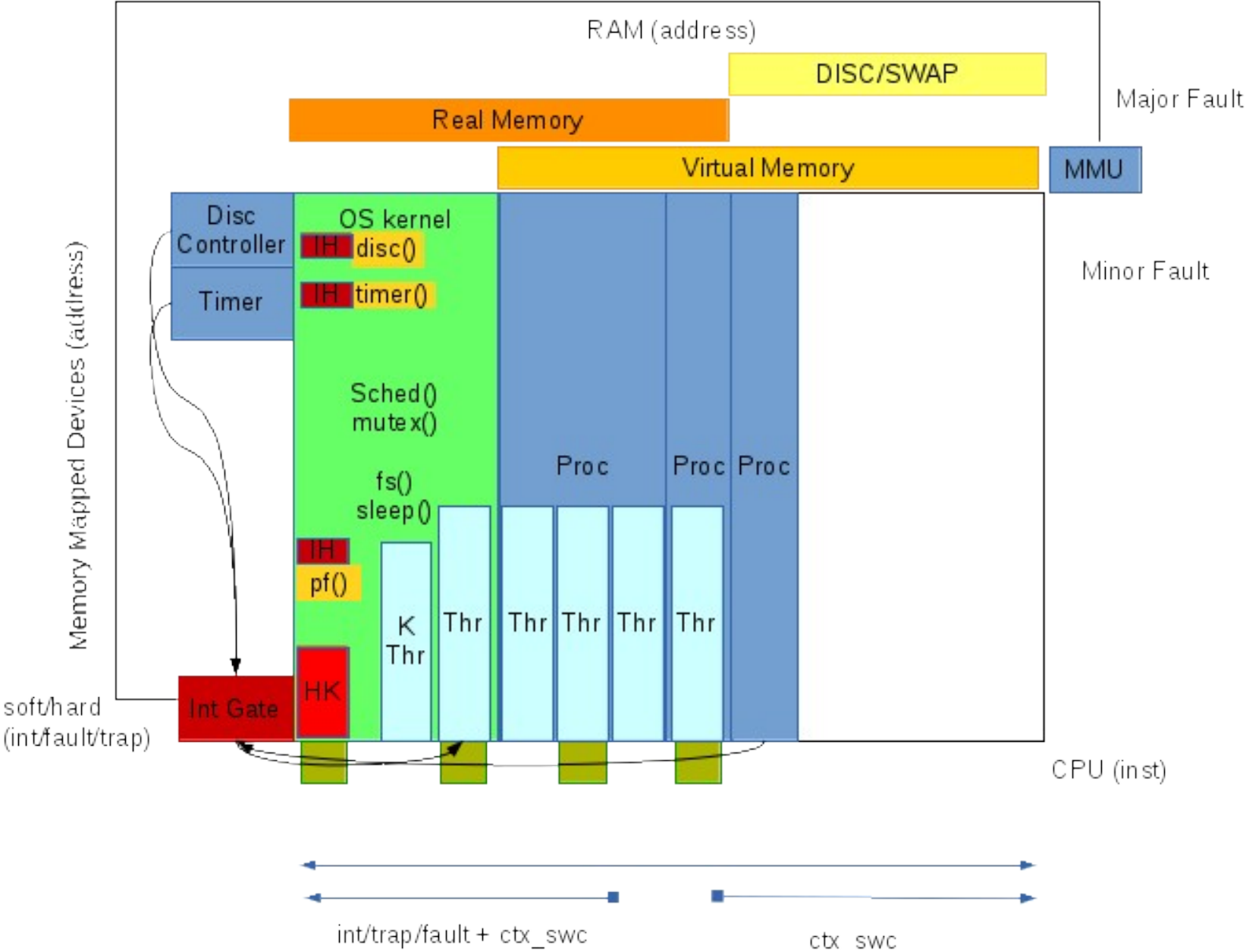
# RT Library?



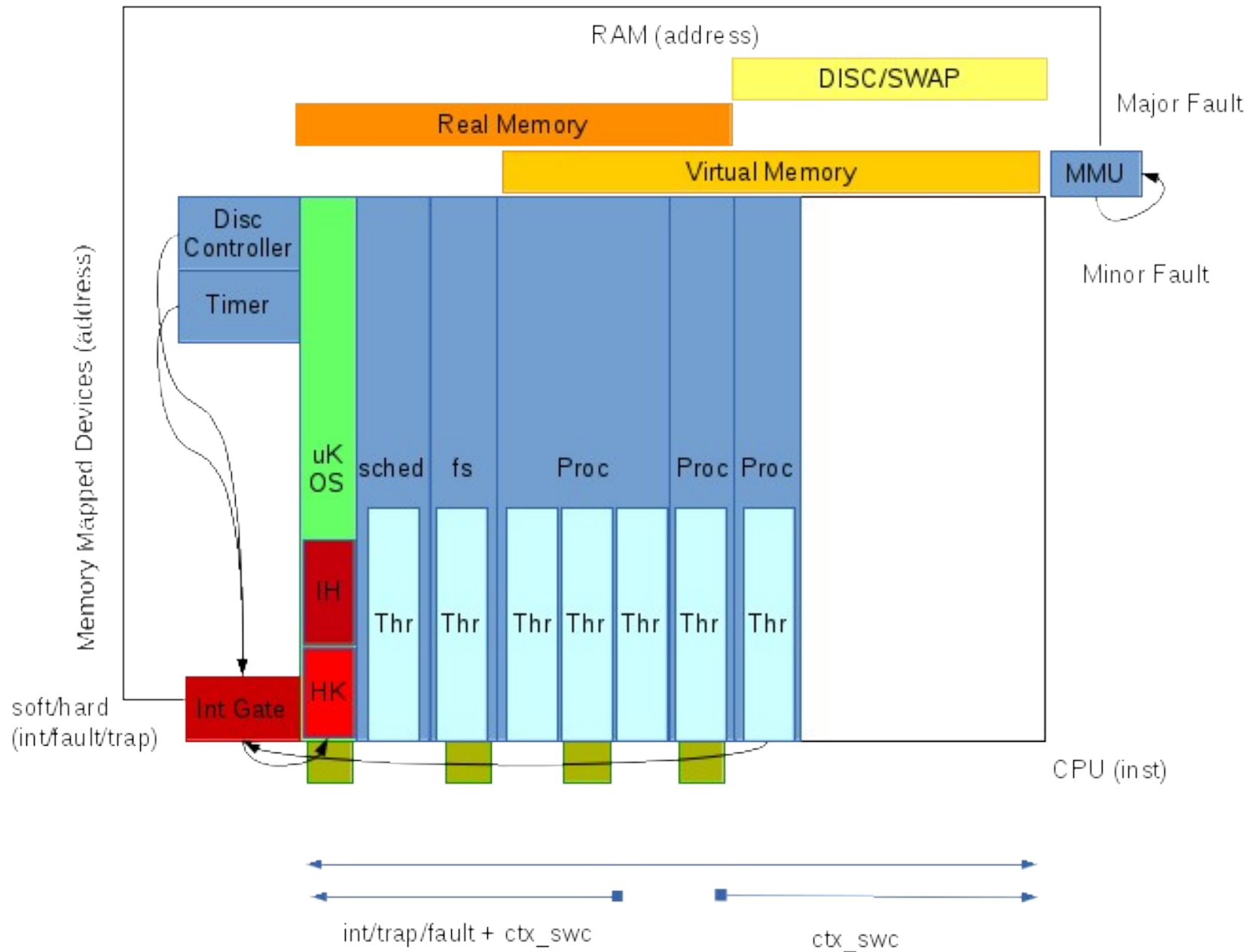
# FreeRTOS



# Linux

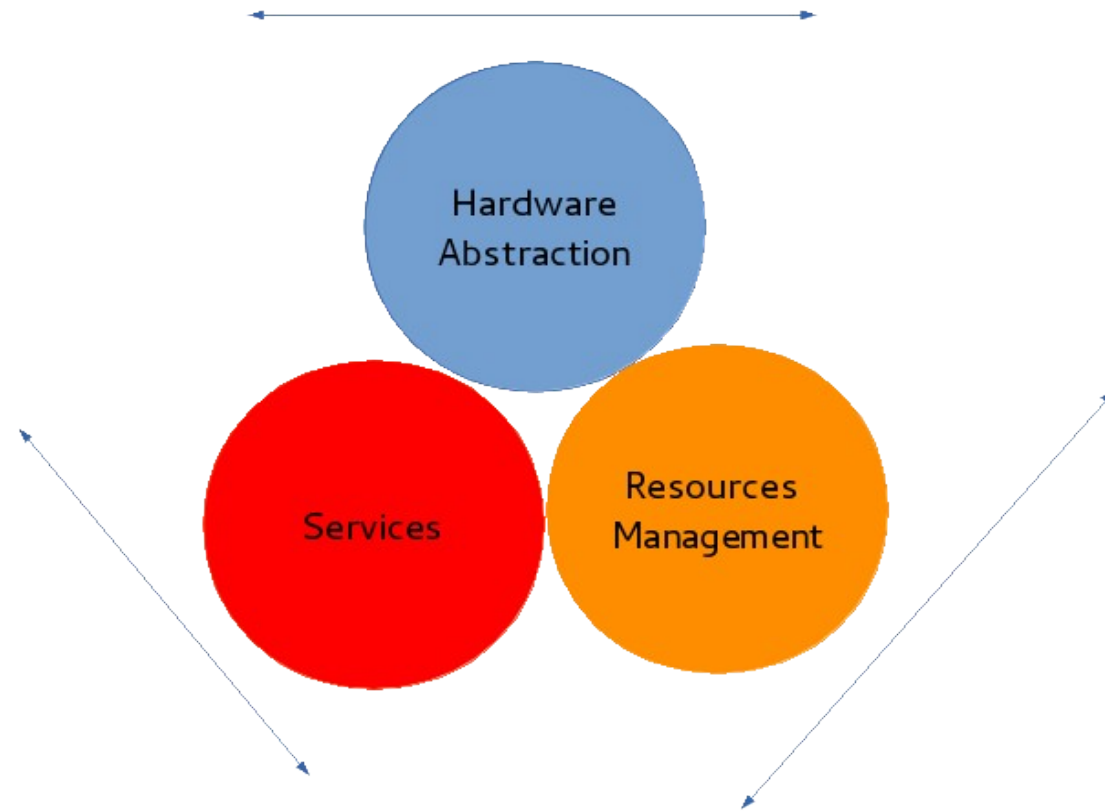


# Neutrino (QNX)



Why is Linux not a  
RTOS?

# Linux is too much flexible



# General Purpose OS

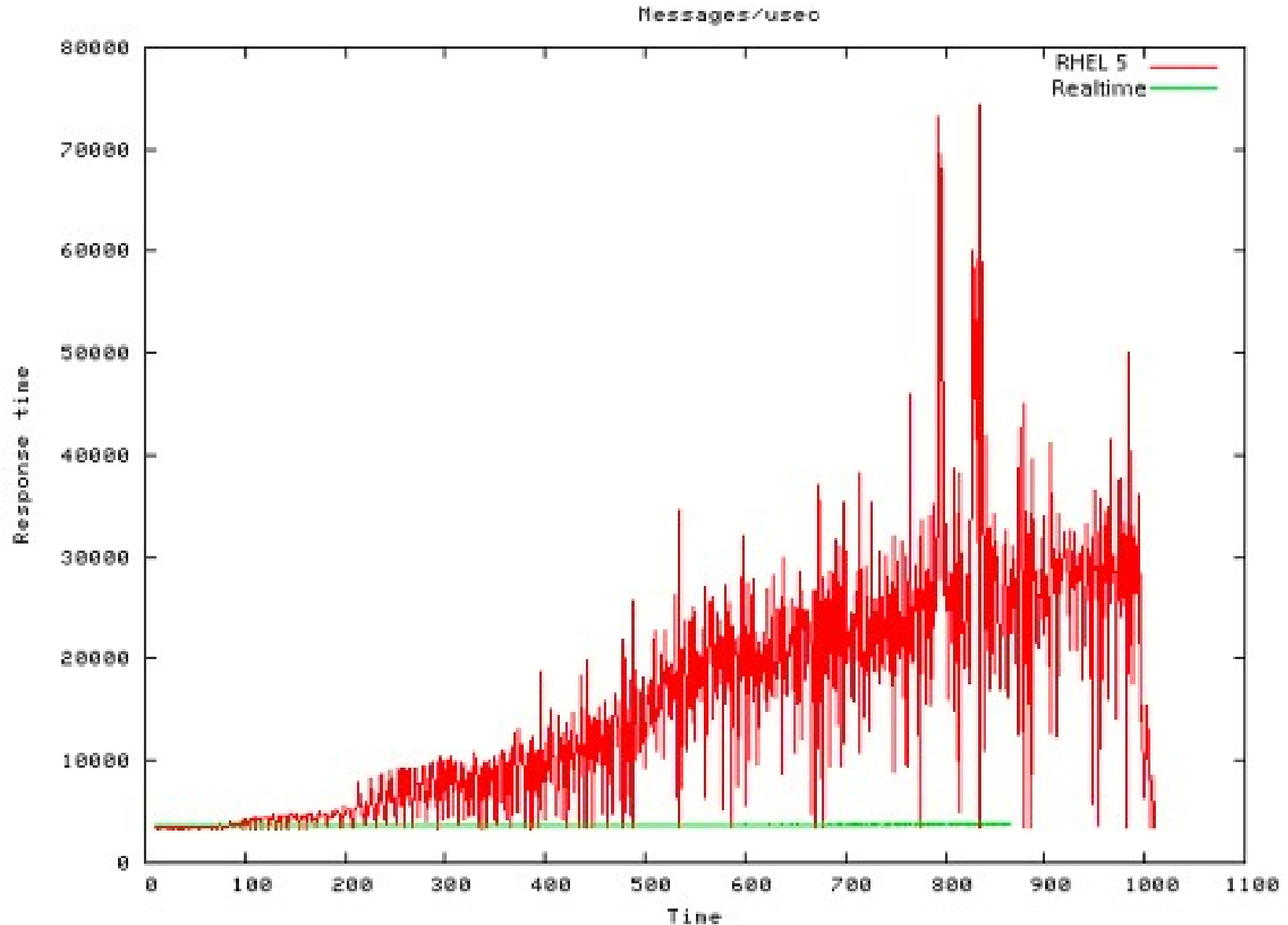
- Linux not is a RTOS by conception
- High throughput rather than high determinism.
  - High latency
- But has:
  - Static priority sched: FIFO and RR
  - Dynamic priority sched: DEADLINE (EDF like).



# Linux as a RTOS?

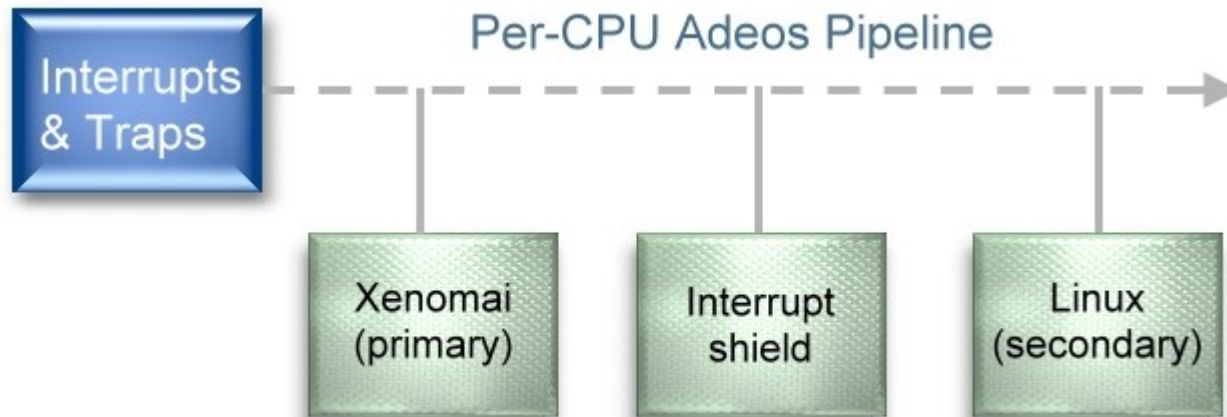
- Why?
- Hardware support:
  - ARM, MIPS, POWERPC, intel, TILE 64, ADBF...
- Services:
  - FS, Network Stack...
- Money:
  - Rich company X needs a low latency OS for stock exchange operations

# How Red Hat shows your Real-Time Linux



# Real-Time Linux: Past, Present, Possible Futures...

RTAI and Xenomai



RT Task on Xenomai

+

Non-RT on Linux...

=

) : )

<- this is a bipolar emoticon

**PREEMPT-RT**

- latency  
+ control over OS tasks

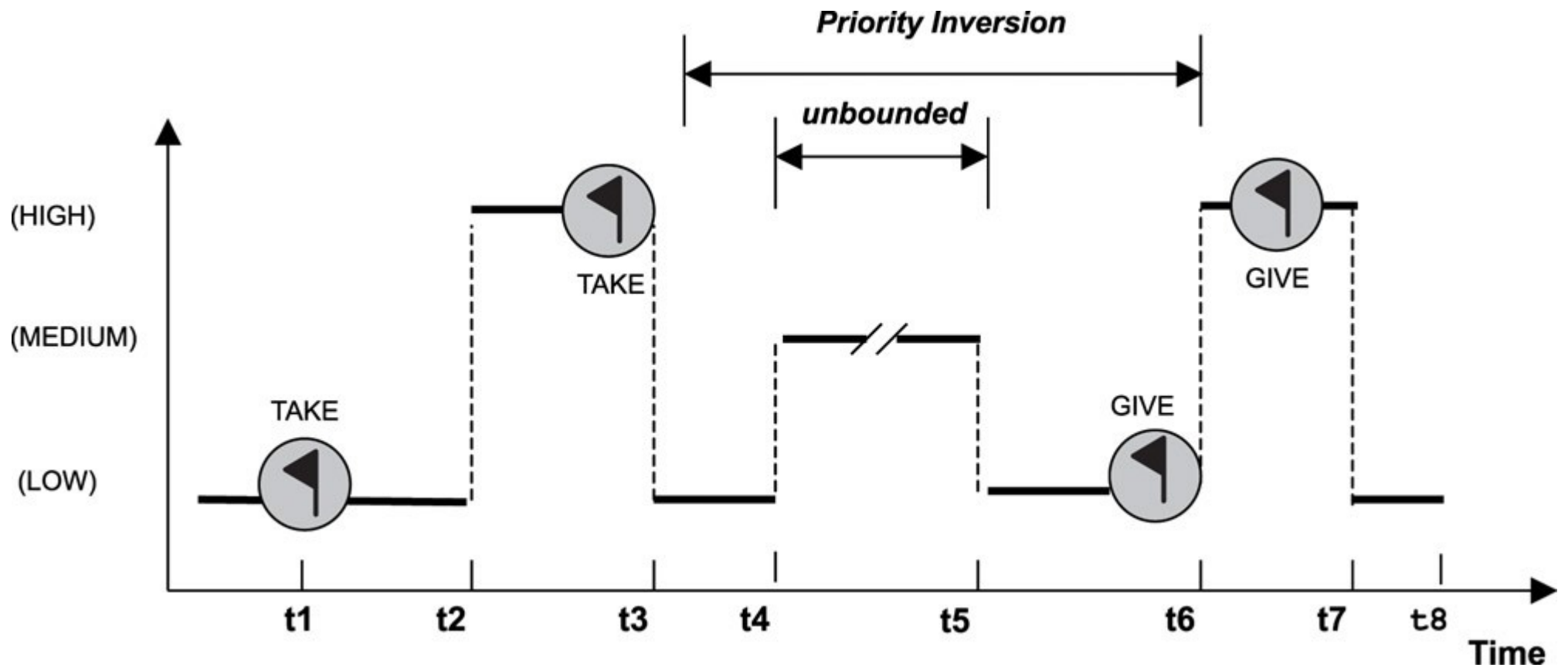


# Sources of latency

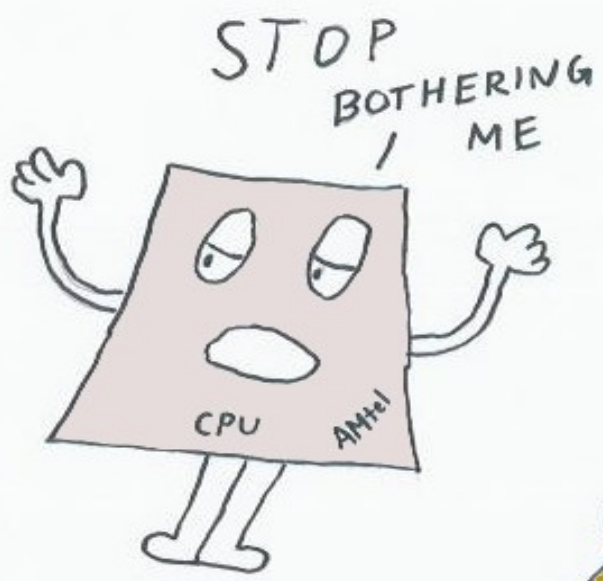
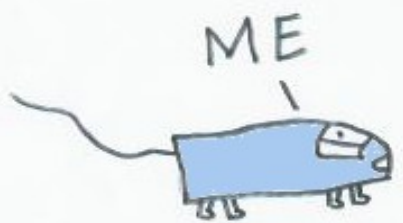
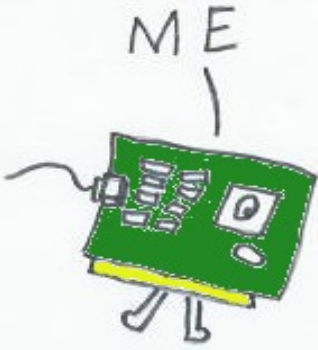
- IRQ Handling
  - less IRQs off latency
  - Softirq
- Preempt off
  - spinlocks

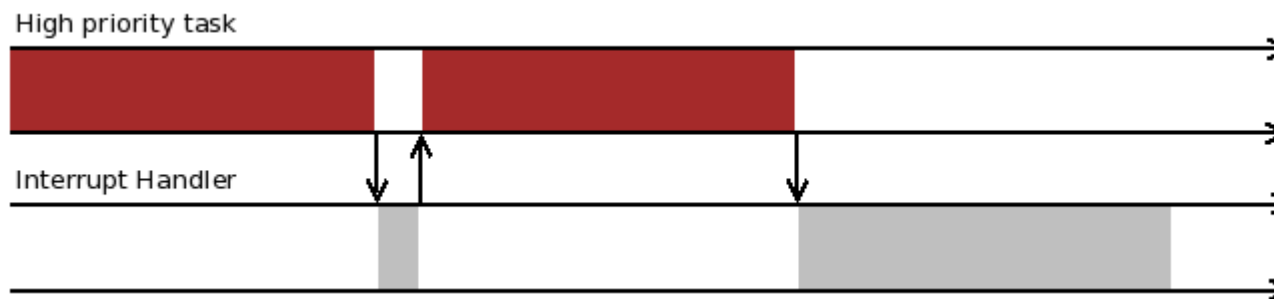
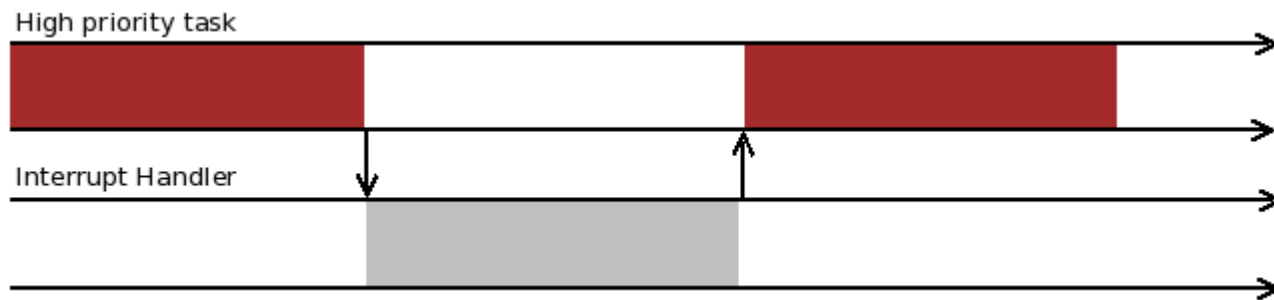
Preemptive lock

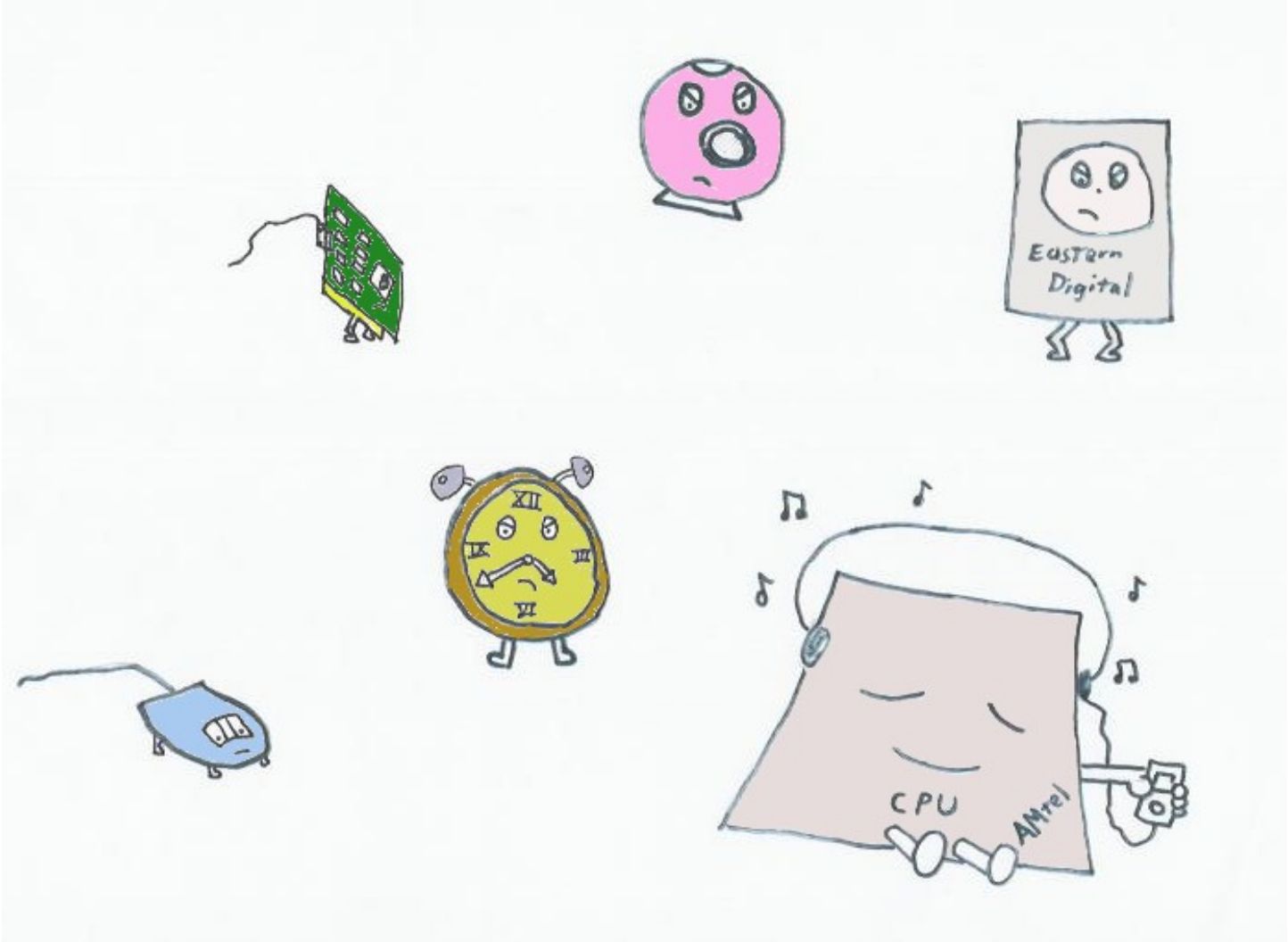
# Priority inversion problems...



# IRQ Handling



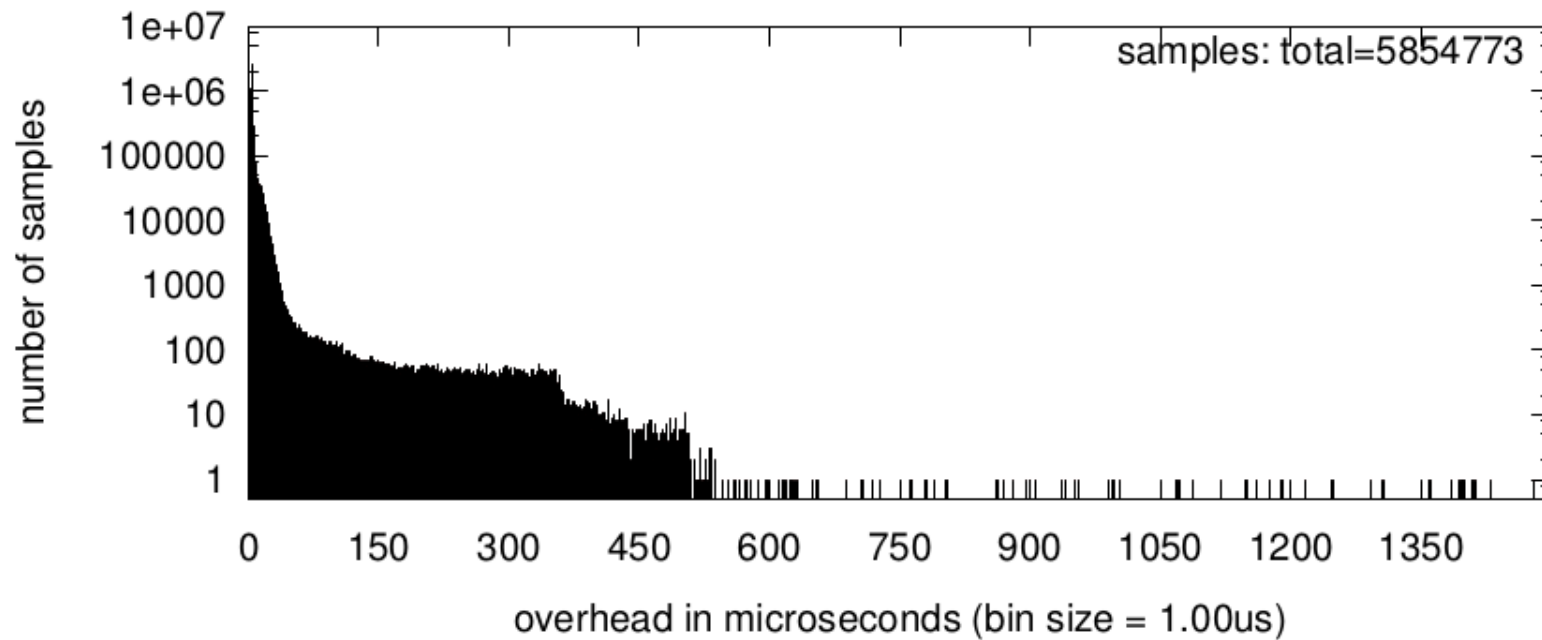




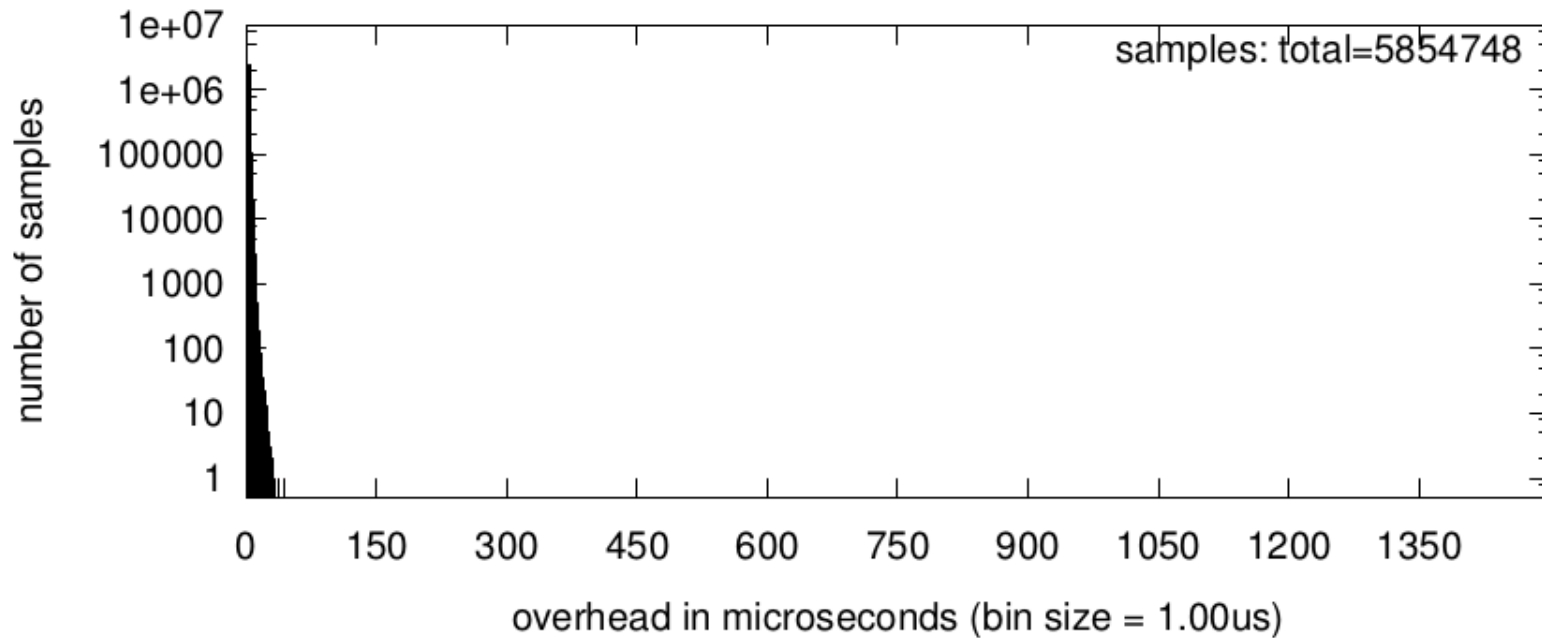


How good is PREEMPT\_RT Today?

Linux 3.8.13: scheduling latency (IO-bound bg tasks)  
min=1.85us max=5464.07us avg=6.23us median=4.60us stdev=15.91us



Linux 3.8.13 w/ PREEMPT-RT: scheduling latency (IO-bound bg tasks)  
min=1.47us max=44.16us avg=4.12us median=4.07us stdev=0.99us



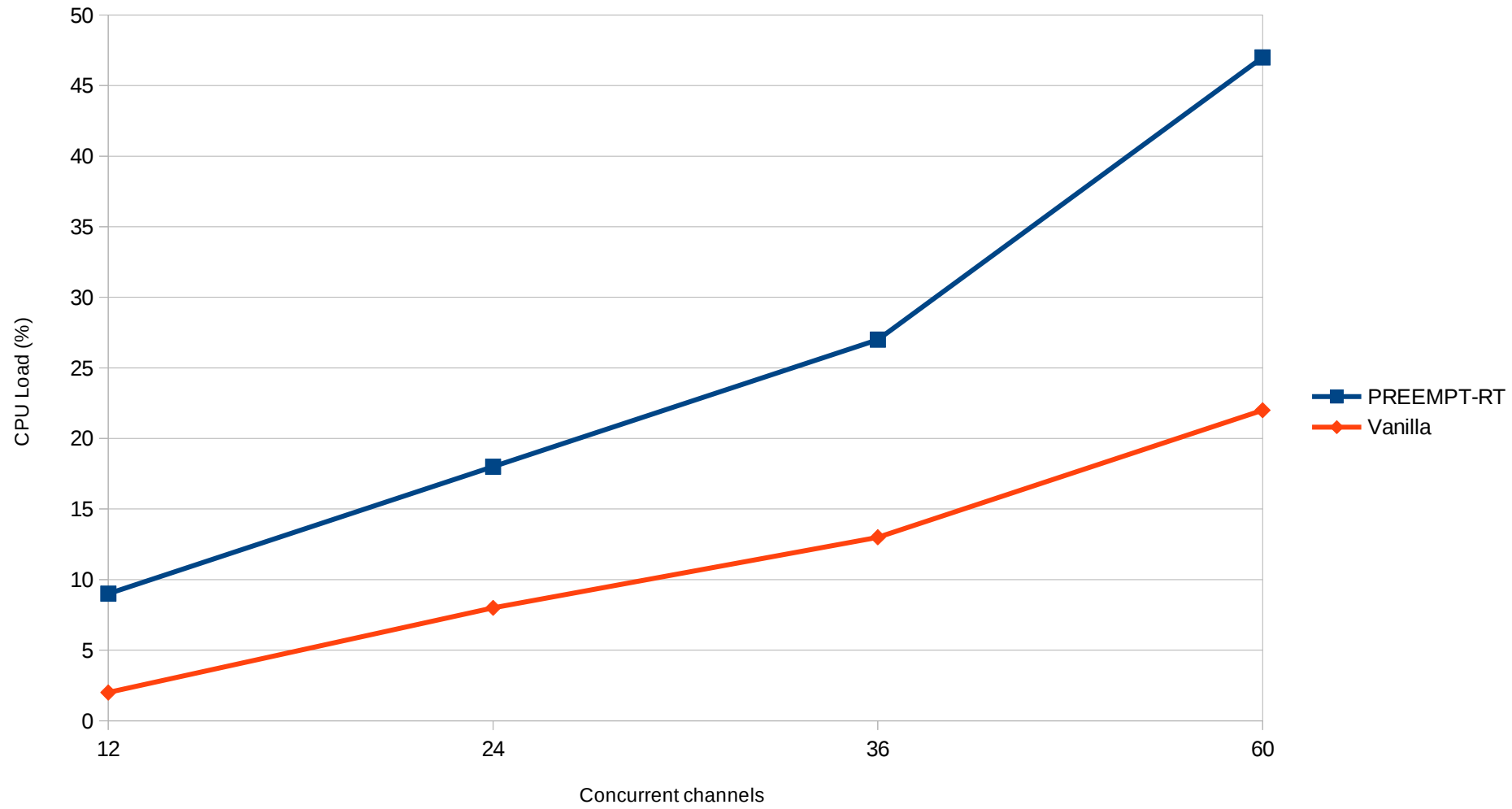
PREMPT-RT is **WONDERFULL!**

Let's use everywhere!!!



This is another bipolar emoticon, but more happy than another...

How expensive is  
**PREEMP-RT?**



LITMUS<sup>RT</sup>

A **testbed** for multiprocessor **scheduling**  
in **Real-Time** systems



(Partitioned|Global|Clustered) EDF

Partitioned Fixed-Priority

PD2 (PFAIR)

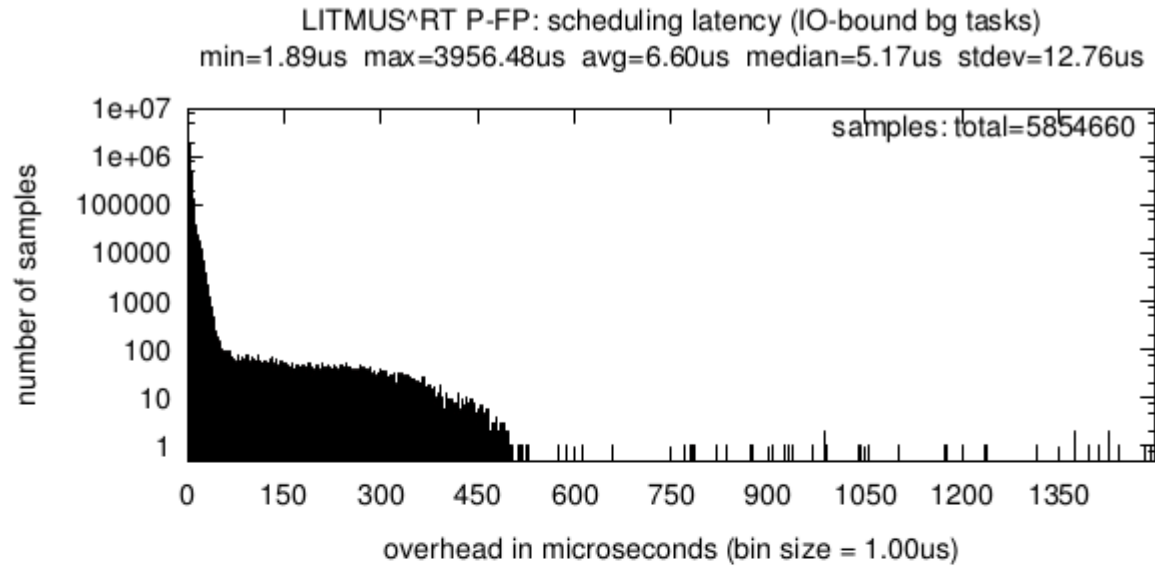
Mutual Exclusion

Interrupt Handling

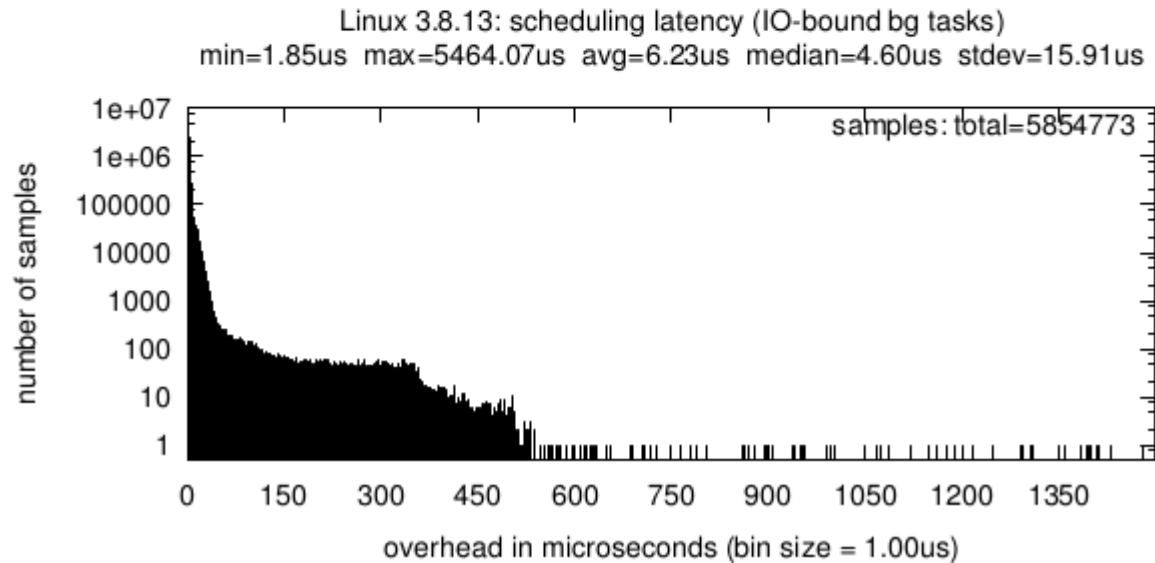
Real-Time GPU Management

Tracing Tools

How good is Litmus RT?



(a) LITMUS<sup>RT</sup> with the P-FP scheduler plugin



Litmus RT

=

a way to pick next task  
with a RT API.

# So...

- Operating system are complex software;
  - More complex → harder warranty determinism
- It is not that easy to provide a completely deterministic OS...
  - It is not only about scheduling/locking algorithms.
- Generally, OS has more than one performance metric to be evaluated:
  - and the throughput is highly punish by the determinism.

Questions?