

# CONCEPTES AVANÇATS DE SISTEMES OPERATIUS (CASO)

Facultat d'Informàtica de Barcelona, Dept. d'Arquitectura de Computadors, curs 2012/2013 – 1Q

## Pràctiques de laboratori

## Temps real - Xenomai

### Material

La vostra instal·lació de Linux.

Els codi de Linux instal·lat a `/usr/src/linux`

Com a la pràctica anterior, si ho veieu necessari, podeu instal·lar-vos una màquina virtual amb Linux. Us pot anar bé per evitar reboots i possibles pèrdues d'informació.

### Entorn

Per a realitzar aquesta pràctica tenim dues alternatives:

1. Compilar el kernel amb Xenomai i realitzar els exercicis
2. Instal·lar un kernel precompilat amb Xenomai i realitzar els mateixos exercicis

### ALTERNATIVA 1: Compilació del kernel amb Xenomai

Xenomai ([www.xenomai.org](http://www.xenomai.org), Xenomai 2.6.1 (stable)) proporciona uns fitxers de modificació del kernel en forma de *patches*. Això fa que haguem de treballar amb versions concretes del kernel de Linux. Per exemple, la versió actual de Xenomai incorpora els següents *patches*:

- `xenomai-2.6.1/ksrc/arch/x86/patches/adeos-ipipe-2.6.37.6-x86-2.9-02.patch` # pel kernel 2.6.37.6
- `xenomai-2.6.1/ksrc/arch/x86/patches/adeos-ipipe-2.6.38.8-x86-2.11-01.patch` # pel kernel 2.6.38.8
- `xenomai-2.6.1/ksrc/arch/x86/patches/ipipe-core-3.2.21-x86-1.patch` # pel kernel 3.2.21

Si la vostra instal·lació de Linux ja té un kernel 3.2, haureu de baixar els fonts del kernel 3.2.21:

- <http://www.kernel.org/pub/linux/kernel/v3.0/linux-3.2.21.tar.bz2>

Si la vostra instal·lació de Linux té un kernel 2.6, és més recomanable usar els fonts del kernel 2.6, que podeu obtenir d'aquí:

- <http://www.kernel.org/pub/linux/kernel/v2.6/> # trieu 2.6.38.8 o bé 2.6.37.6

Podeu seguir les instruccions per compilar el kernel amb Xenomai, d'aquí: [http://www.xenomai.org/index.php/Xenomai\\_quick\\_build\\_guide](http://www.xenomai.org/index.php/Xenomai_quick_build_guide)

Observeu que haureu de canviar `xenomai-2.4.4` per la darrera versió `xenomai-2.6.1` i la usar la versió

del kernel que hagueu seleccionat abans.

Per aconseguir el fitxer de configuració (.config) del kernel per defecte, recordeu que podeu fer servir l'eina *make* amb `i386_defconfig` o bé `x86_64_defconfig`, que generaran configuracions correctes preparades per córrer a la majoria de màquines Intel. Si aconsegiu així la configuració del kernel, activeu/desactiveu a continuació les següents opcions:

- Desactivar "Processor type and features => Tickless System (Dynamic Ticks)"
- Desactivar "Power management and ACPI options => ACPI (...) Support"
- Desactivar "Power management and ACPI options => CPU Frequency scaling => CPU Frequency scaling"
- Desactivar "Power management and ACPI options => Cpuidle Driver for Intel Processors"
- Activar "Device Drivers => Generic Driver Options => Maintain a devtmpfs filesystem to mount at /dev"
- Activar "Device Drivers => Generic Driver Options => Automount devtmpfs at /dev... "
- Activar "File systems => The Extended 4 (ext4) filesystem"
- Activar "File systems => FUSE (Filesystem in Userspace) support"
- Activar "Real-time Subsystem => Extra scheduling classes"
- Activar "Real-time Subsystem => Temporal partitioning"
- Activar "Real-time Subsystem => Sporadic scheduling"
- Activar totes les opcions a "Real-time Subsystem => Drivers => Real-time IPC drivers"

A continuació compileu el kernel, els mòduls (si n'hi ha), i instal·lar els mòduls i el kernel:

- `make` # per compilar el nucli del kernel
- `make modules` # per compilar els mòduls
- `make modules_install` # per instal·lar els mòduls a /lib/modules
- `make install` # per instal·lar el kernel a /boot

I ara podeu rebotar per començar a treballar amb Xenomai.

## **ALTERNATIVA 2: Instal·lació d'un kernel precompilat amb Xenomai**

Podeu baixar els binaris del kernel 3.2.21 ja precompilat amb Xenomai-2.6.1, incloent les utilitats de Xenomai, de la pàgina web de l'assignatura:

<http://docencia.ac.upc.edu/FIB/grau/CASO/laboratori/linux-3.2.21-xenomai-2.6.1.tar.bz2>

En aquesta instal·lació trobareu:

- `/boot/vmlinuz-3.2.21-xenomai-2.6.1` # el kernel
- `/lib/modules/3.2.21-ipipe/` # el directori amb els mòduls del kernel

- ☑ /usr/xenomai/                   # el directori amb les utilitats de Xenomai
- ☑ /usr/xenotest.tgz               # el codi font de l'aplicació que farem servir de test

Desplegueu el fitxer a l'arrel del vostre sistema i canvieu la configuració del boot loader, per què pogueu triar el nou kernel amb Xenomai: vmlinuz-3.2.21-xenomai-2.6.1.

Rebooteu per començar a usar Xenomai.

## Exercicis sobre Xenomai

Desplegueu el paquet de Xenomai-2.6.1 ([www.xenomai.org](http://www.xenomai.org), Xenomai 2.6.1 (stable)).

- 1) Entreu en el directori xenomai-2.6.1/examples/native i mireu el programa trivial-periodic.c. És l'exemple que vam veure a classe. Compileu-lo i executeu-lo. Comproveu que el "time since last turn" que indica és prou acurat, al voltant d'un segon.
- 2) Useu la comanda real-time ps ('rtps'), que trobareu instal·lada a /usr/xenomai/sbin/rtps per comprovar que trivial-periodic s'executa en el domini de temps real.
- 3) Useu la comanda ./sigdebug per comprovar que Xenomai informa als processos que surten del domini de temps real per culpa d'una crida a sistema.
- 4) Canvieu la crida printf ("Switched to secondary mode\n"); per un pid=getpid(); Així podreu comprovar que hi ha crides a sistema que no fan sortir el procés de temps real. Busqueu a la documentació de Xenomai com podem reduir les transicions cap a fora de temps real per a les aplicacions.
- 5) Veieu el programa rtpint.c. Aquest programa usa el servei rt\_printf de Xenomai, que permet fer entrada/sortida sense que el procés deixi d'estar en mode de temps real. Observeu que la primera invocació de rt\_printf encara treu el procés de temps real, però les següents ja no.
- 6) Desplegueu el paquet xenotest.tgz. Mireu el programa format pels fitxers periodic-thread.c, periodic-tasks.c i task\_bodies.c, i enteneu-lo per veure què fa.

Compileu la part d'usuari del programa: make -f Makefile.userspace, i executeu-la:

```
$ ./periodic-thread
```

El programa crea 3 threads i cadascun executa una tasca periòdica. Les dues primeres tasques només imprimeixen números (1 i 2) quan els toca executar-se. La tercera tasca comprova que s'hagi despertat a temps, imprimint el temps que ha passat des de la darrera invocació, que habitualment són 70 milisegons.

## Entrega, entre el dimecres 5 i el divendres 14 de desembre

A la classe de laboratori, en forma de demo.