

Disseny de Sistemes Operatius

Facultat d'Informàtica de Barcelona - FIB

Enginyeria Informàtica

PRÀCTICA 2: Disseny, implementació i avaluació d'entorns operatius

1. Objectius de la pràctica

L'objectiu de la segona pràctica és analitzar i criticar experimentalment, sempre que sigui possible, l'estat de l'art del disseny de sistemes operatius actuals. Es tracta d'entendre l'evolució i la situació actual de les abstraccions que ofereix un sistema operatiu, les llibreries de threads, la interfície dels (micro)kernels, els objectes interns i externs i la seva funcionalitat, el desenvolupament d'aplicacions i servidors directament sobre les noves interfícies, el pas de missatges, la gestió de recursos, etc.

Tenim diversos entorns de treball per a la realització d'aquesta pràctica. Ja coneixeu Solaris (sobre SPARC) i Linux i el microkernel de Mach sobre processadors Pentium d'Intel. També podem pensar en treballar en entorns Windows NT, IRIX, AIX...

2. Parts del treball a realitzar

La segona pràctica comprèn (Aquests punts depenen de la pràctica concreta que es faci):

- l'estudi de l'estat de l'art d'un dels temes següents (o part del tema), basant-se principalment en cercar i llegir uns quants articles que parlin del tema i fent un petit resum, i/o resseguint el codi font del kernel;
- el disseny i la realització d'un joc de proves que serveixi per comprovar els dubtes sobre el funcionament del que estem estudiant i que només llegint la documentació puguin sorgir, i per avaluar el seu rendiment;
- presentació del treball, com a mínim amb una documentació escrita, però si es possible, també oralment davant de la classe.

La documentació resultat d'aquesta pràctica, sempre que assoleixi un nivell mínim de qualitat, serà pública i s'incorporarà com a documentació de l'assignatura.

3. Proposta de temes

T1. Programació multíflux

Estudi de la situació actual de les abstraccions de threads de kernel i threads d'usuari. Veure la implementació d'algunes d'aquestes llibreries de threads, les seves funcionalitats, el seu model de programació i el seu mapeig sobre els objectes del kernel on corren, i opinar sobre la relació entre el disseny i el rendiment que poden donar a les aplicacions paral·leles. Proposem escollir dos o tres llibreries d'entre: Pthreads, Cthreads, Qthreads, LWP (Sun, HP)...

Possibles temes:

Estudi d'interfícies de fluxos a nivell de kernel i d'usuari en nous kernels i llibreries de threads.

Propostes de depuració i traceig de programes multíflux. Problemes que apareixen i disseny d'una solució.

Avaluació de la seva eficiència mesurant el cost de creació de les abstraccions d'usuari i sincronització entre threads. Dissenyar i implementar alguns benchmarks que us ajudin a realitzar les mesures.

T2. L'eficiència en el disseny de servidors

Es pot fer l'anàlisi de l'eficiència deguda als diferents tipus de disseny de servidors. Per exemple, servidors seqüencials, servidors replicats amb processos i servidors paral·lels amb fluxos. Es contemplarà també la qüestió de la creació estàtica o dinàmica de les instanciacions dels serveis. S'intentarà avaluar el rendiment que aporta cadascun d'ells i l'àmbit més adequat pel seu ús.

Es podrien estudiar els rendiments associats a treballar en un entorn monolític o microkernel, amb arquitectura centralitzada o distribuïda, avaluant els diferents mecanismes de comunicacions que tinguem disponibles segons el sistema: sockets, ports, Winsocks NT, etc.

T3. Sistemes de fitxers

Analitzar l'estructura i funcionalitats que tenen els sistemes de fitxers avançats i com s'ha evolucionat des de sistemes de fitxers senzills (com el de System V) fins arribar a sistemes de fitxers que permeten gestió de volums, alta disponibilitat, ràpida recuperació, que són tolerants a fallades, etc., com poden ser el Ext3 de Linux, el AdvFS de Compaq, NTFS de Window2000 o el XFS d'IRIX.

Determinar si aquests nous serveis que s'afegeixen als sistemes de fitxers són totalment transparents a l'usuari, o bé també s'introdueixen noves interfícies d'accés i/o gestió del sistema de fitxers, de forma que amb aquestes s'aconsegueixi obtenir un rendiment més alt del sistema de fitxers.

Es poden posar en pràctica les funcionalitats dels nous sistemes de fitxers preparant algun benchmark sintètic que estressi alguna de les aportacions d'aquests sistemes de fitxers i permeti comparar el seu rendiment amb el d'un sistema de fitxers tradicional.

T4. Gestió dels recursos

Aprofundir en el disseny de les formes d'accés als recursos (dispositius). Estudiar l'evolució des de UNIX fins a les noves abstraccions dels microkernels: quins canvis s'han produït en el control d'accés als recursos i en les polítiques de gestió per part dels usuaris.

Fer un ventall de les possibilitats que dona el kernel i/o el subsistema d'entrada/sortida (discos, xarxa, multimèdia). Determineu qui s'encarrega de la feina, si és possible treballar en algun d'aquests modes d'entrada/sortida: síncron/asíncron, directe/indirecte, raw/cooked, seqüencial/paral·lel... Estudiar com es pot dur a terme l'avaluació del rendiment de l'entrada/sortida en els sistemes actuals i preparar algun benchmark.

Estat de l'art de la parametrització dinàmica que permeten els sistemes operatius actuals. Es tracta de veure quina informació dona el sistema sobre l'estat de si mateix, de les abstraccions que gestiona, del subsistema i dels servidors. Avaluar els efectes sobre el rendiment de l'ús adequat de la sintonització. Es a dir, comparar amb un programa d'exemple que la modificació d'algun paràmetre pot ser útil a les aplicacions per tal d'aconseguir una execució més eficient. Fer les mateixes proves amb un benchmark més complet.

Estudiar la gestió que els sistemes operatius multiprocessadors fan dels recursos processador i memòria, i la planificació de fluxos en aquests entorns d'execució.

T5. Emulació de sistemes / extensibilitat

Es tracta de fer un estudi del mapeig de les abstraccions dels sistemes operatius clàssics sobre les noves abstraccions del microkernel. Pot incloure, a més a més, el tractament del TRAP, la redirecció de les peticions de servei cap al subsistema, la gestió dels espais d'adreces...

Quins aspectes rellevants en el disseny, fiabilitat i eficiència provenen de la implementació de l'emulació mitjançant llibreries d'emulació i/o servidors. Limitacions i evolució.

Dependències del subsistema per part dels processos, control des del subsistema dels seus processos, gestió d'excepcions, dels recursos, de la càrrega de processos...

Quines facilitats i quines necessitats demana la possible coexistència de subsistemes emulant diferents sistemes operatius sobre un mateix microkernel.

Estudi i experiències de treball en un entorn *embedded* basat en processador Intel i sistema Windows.

Visió actual de l'extensibilitat. Formes de contruir les imatges executables dels processos mitjançant mòduls i enllaços dinàmics. Implementació de serveis extensibles.

T6. Tècniques de tracejat de processos

Estudi de la implementació de les tècniques tradicionals de tracejat de fluxos en execució amb crides a sistema especials (ptrace de UNIX), i com es pot fer a partir de les eines que aporten els nous microkernels.

Actualment els processos poden ser també vistos com a fitxers, i podem utilitzar les crides a sistema tradicionals sobre el sistema de fitxers /proc per a controlar-los (open, close, read, write, ioctl).

Determinar les possibilitats d'obtenir informació i de controlar els fluxos i la memòria de les aplicacions quan es treballa sobre alguns d'aquests sistemes: Mach 3.0, Mach 2.5, Linux, IRIX, Windows NT, Solaris.

T7. Gestors externs de memòria / Memòria Virtual Distribuïda

Les polítiques de gestió de memòria a alt nivell es situen fora del kernel. Aprofundiment en el disseny dels gestors externs, protocols de comunicació, funcionalitats per les aplicacions d'usuari.

Proposta d'implementació d'un paginador extern, interfície i funcionament.

Proposta de disseny d'un gestor extern de memòria física que donés, amb el suport dels mecanismes adients dins el kernel, una abstracció memory set de funcionalitat anàloga al processor set.

Estudi de tècniques de Memòria Virtual Distribuïda (DSM) i dels mecanismes de manteniment de la coherència.

Avaluació del rendiment de la gestió de la memòria. Fer uns benchmarks amb diferents patrons d'accés a posicions de memòria i fer algunes mesures; per exemple: traceig de les fallades de pàgina i sobrecàrrega en els temps d'execució.

4. Accés a documentació

Podeu trobar els articles que descriuen el disseny dels aspectes relacionats amb microkernels a les col·leccions de la Universitat de Carnegie Mellon, la Universitat de Utah i a la Open Software Foundation. Hem fet una còpia local al FTP del Departament d'Arquitectura de Computadors (<ftp.ac.upc.es/pub/archives/gso/mach.CMU>, mach.UTAH i mach.OSF). Són

accessibles i consultables més fàcilment (hi ha abstracts que resumeixen els continguts dels articles) anant als repositoris d'articles i documentació dels diversos sistemes des de la pàgina de DSO (<http://www.ac.upc.es/docencia/FIB/DSO>)

Podeu demanar un préstec dels articles en paper (ja que molts estan en format *postscript*) d'aquestes col·leccions.

D'altra banda, podeu trobar el codi font del microkernel Mach 3.0 al directori /system/ de fossas.ac.upc.es (per exemple, els fonts del kernel són a /system/gsomk_ad3/src/mach_kernel), i els de la darrera versió de Linux a marsa.ac.upc.es.

5. Entrega de la pràctica

L'entrega de la pràctica consistirà en un informe del treball realitzat amb l'enunciat concret pel grup acordat amb el professor de laboratori, el disseny fet, i la realització de les proves. Es pot entregar una versió en paper, però es demana que hi hagi una versió en ASCII o en HTML que s'entregui per correu electrònic, juntant els fitxers de l'informe i els codis desenvolupats en un paquet *shar*.