



Departament d'Arquitectura  
de Computadors

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# Conceptes Avançats de Sistemes Operatius

Facultat d'Informàtica de Barcelona  
Dept. d'Arquitectura de Computadors

Curs 2018/19 Q2

Mach

UNIVERSITAT POLITÈCNICA DE CATALUNYA  
BARCELONATECH

Facultat d'Informàtica de Barcelona



# Interfície Mach

## Abstraccions

Host

Processor / Processor set

Device

Task

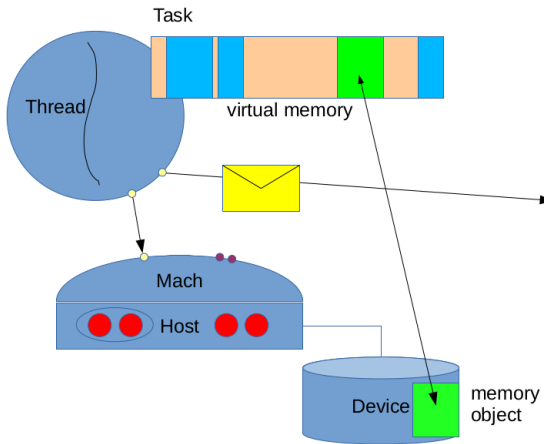
Thread

VM (Virtual Memory)

Port / Port Set

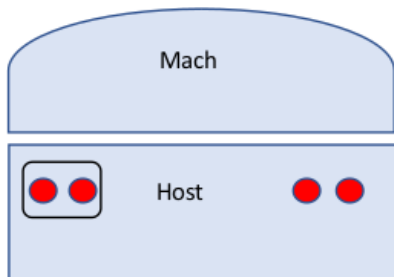
Memory Object

Message



# Host interface

- ▶ Obtenció d'informació
  - ▶ Bàsica: processadors, tipus i memòria disponible
  - ▶ Càrrega instantània (uptime, top)
  - ▶ Planificació: timeout, quantum mínims (en ms.)
  - ▶ Versió del kernel
  - ▶ Data i hora

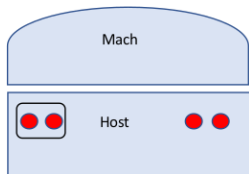


# Host interface

- ▶ Canvi de la informació
  - ▶ Ajustar data i hora
- ▶ Interactuar amb el host
  - ▶ Reboot
- ▶ Aconseguir els ports privilegiats del host
  - ▶ Ofert pels servidors de sistema(?)

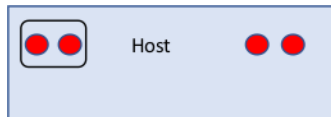
```
int get_privileged_ports(  
    &host_privileged_port,  
    &device_privileged_port  
);
```

- ▶ Només accessible a root



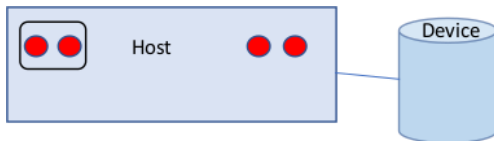
# Processor interface

- ▶ Aconseguir informació
  - ▶ Tipus, si està funcionant, slot, master
  - ▶ Processor set al qual està vinculat
- ▶ Parar i engegar processadors
- ▶ Obtenir la llista de processor sets
- ▶ Obtenir accés a processor sets per nom
- ▶ Assignar i desassignar processadors a processor sets
- ▶ Crear i destruir processor sets
- ▶ Obtenir informació del processor set
  - ▶ Tasks associades
  - ▶ Threads associats



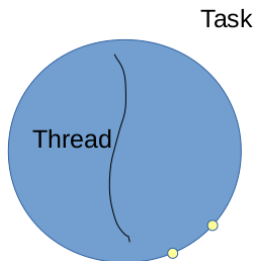
# Device interface

- ▶ Open / close / read / write / map
- ▶ Asynchronous open / read / write
- ▶ Status
- ▶ Filter



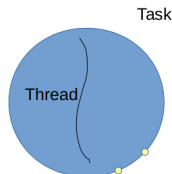
# Task interface

- ▶ Creació / destrucció
- ▶ Aturar / continuar
- ▶ Canviar prioritat
- ▶ Aconseguir informació
  - ▶ comptador d'aturades, prioritat, mides de memòria
  - ▶ temps d'execució per threads vius i acabats
  - ▶ llista de threads
- ▶ `mach_task_self (3.0)` / `task_self (2.5)`
- ▶ get / set special ports
  - ▶ bootstrap, exception, kernel



# Thread interface

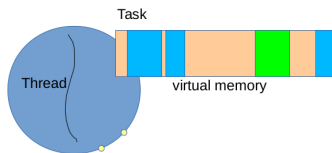
- ▶ Creació / destrucció / consultar/assignar registres
- ▶ Aturar / continuar / canvi de context
- ▶ Canviar prioritat, política de planificació i assignació a processadors
- ▶ Aconseguir informació
  - ▶ comptador d'aturades, política de planificació, prioritat, estat
  - ▶ temps d'execució, ús de cpu
- ▶ `mach_thread_self` (3.0) / `thread_self` (2.5)
- ▶ get / set special ports
  - ▶ exception, kernel





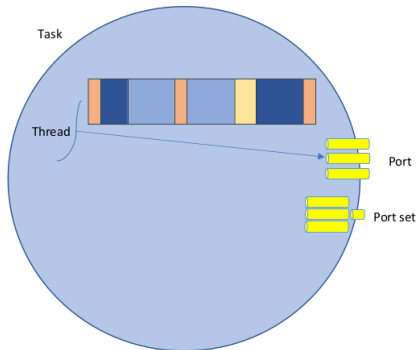
# Virtual Memory interface

- ▶ Demanar i alliberar memòria anònima
- ▶ Còpia entre regions de memòria
- ▶ Mapejar memory objects
- ▶ Fixar la memòria virtual a la física
- ▶ Obtenir informació sobre les regions de memòria

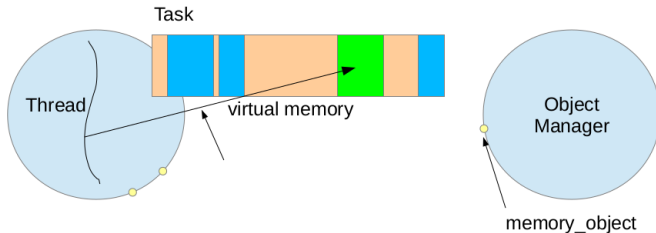


# Port interface

- ▶ Creació / destrucció de ports i port sets
- ▶ Canvi de drets sobre ports
- ▶ Enviar / rebre
- ▶ Informació sobre un port
- ▶ Comptador de referències
- ▶ Estat
- ▶ Creació / destrucció de port sets
- ▶ Moure ports a port sets

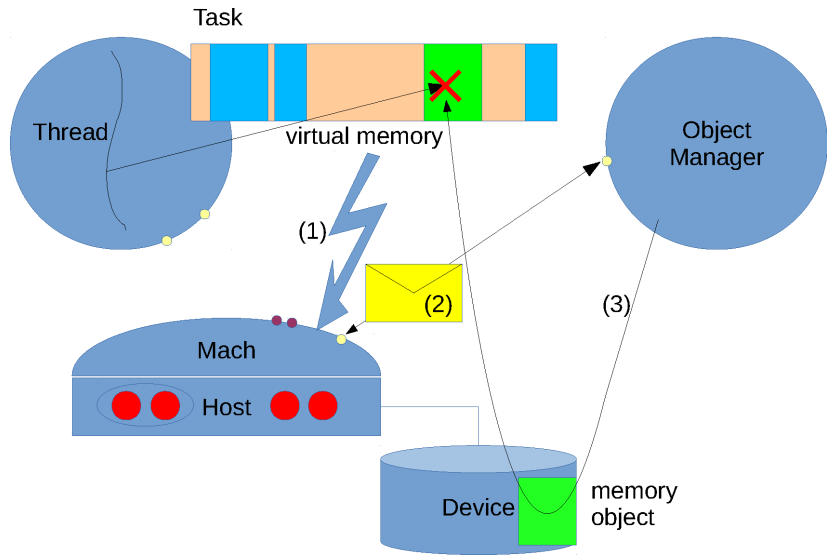


# Memory Object interface



```
kern_return_t vm_map (  
    task_t            target_task,  
    vm_offset_t       address,  
    vm_size_t         size,  
    vm_offset_t       alignment_mask,  
    boolean_t         anywhere,  
    memory_object_t   memory_object,  
    vm_offset_t       offset,  
    boolean_t         copy_or_map,  
    vm_prot_t         cur_protection,  
    vm_prot_t         max_protection,  
    vm_inherit_t      inheritance  
);
```

# Memory Object interface



# Memory Object interface

- (1) Illegal address exception
- (2) `memory_object_init` / `memory_object_ready`  
`memory_object_data_request` → `pagein` /  
`memory_object_data_supply` → `dades`  
`memory_object_data_unavailable` → `zeros`  
`memory_object_data_error` → `segfault`  
`memory_object_data_return` → `pageout`  
`memory_object_data_unlock` → `permettre accès`
- (3) `vm_allocate` / `vm_deallocate`  
`device_read` / `device_write`

# Message interface

- Enviar i rebre missatges a ports

```
mach_msg_return_t mach_msg (  
    mach_msg_header_t*    msg,  
    mach_msg_option_t     option,  
    mach_msg_size_t       send_size,  
    mach_msg_size_t       rcv_size,  
    mach_port_t           rcv_name,  
    mach_msg_timeout_t     timeout,  
    mach_port_t           notify  
);
```

