



Noyau

L4

Mardi 20 juin 2006

Germain BAUVIN
Matthieu MICHAUD
Pierre-Yves ROFES-VERNIS

[bauvin_g]
[michau_m]
[rofes-_p]





Plan

- Intro
- Historique
- La famille L4
- Concepts
 - Espace d'adressage
 - thread et IPC
 - clan and chiefs
 - Unique ID et Flexibilité
- Quelques Projets
 - FIASCO
 - L4::Pistachio
- Webographie
- Conclusion





Intro

"A concept is tolerated inside the microkernel only if moving it outside the kernel, i.e., permitting competing implementations would prevent the implementation of the systems's required functionality."

Liedkte, "On Microkernel Construction", 1995





Historique

- L3
- L4/x86
- L4/Alpha
- L4/MIPS
- Hazelnut





La famille L4

- L4 est une API
 - *μ-noyaux de seconde génération*
- Plusieurs implémentations encore en cours :

Name	CPU	Implementation language	License
<u>Pistachio</u>	Pentium or better, IA64, PowerPC, Alpha, 64-bit MIPS	C++	BSD license
<u>Fiasco</u>	i486 or better, StrongARM, Linux (user-mode L4 emulation)	C++	GPL or commercial
<u>P4</u>	x86, MIPS, PowerPC, ARM	C	commercial
<u>L4 for PowerPC</u>	PowerPC 603e	C	Not yet available





Quelques Concepts





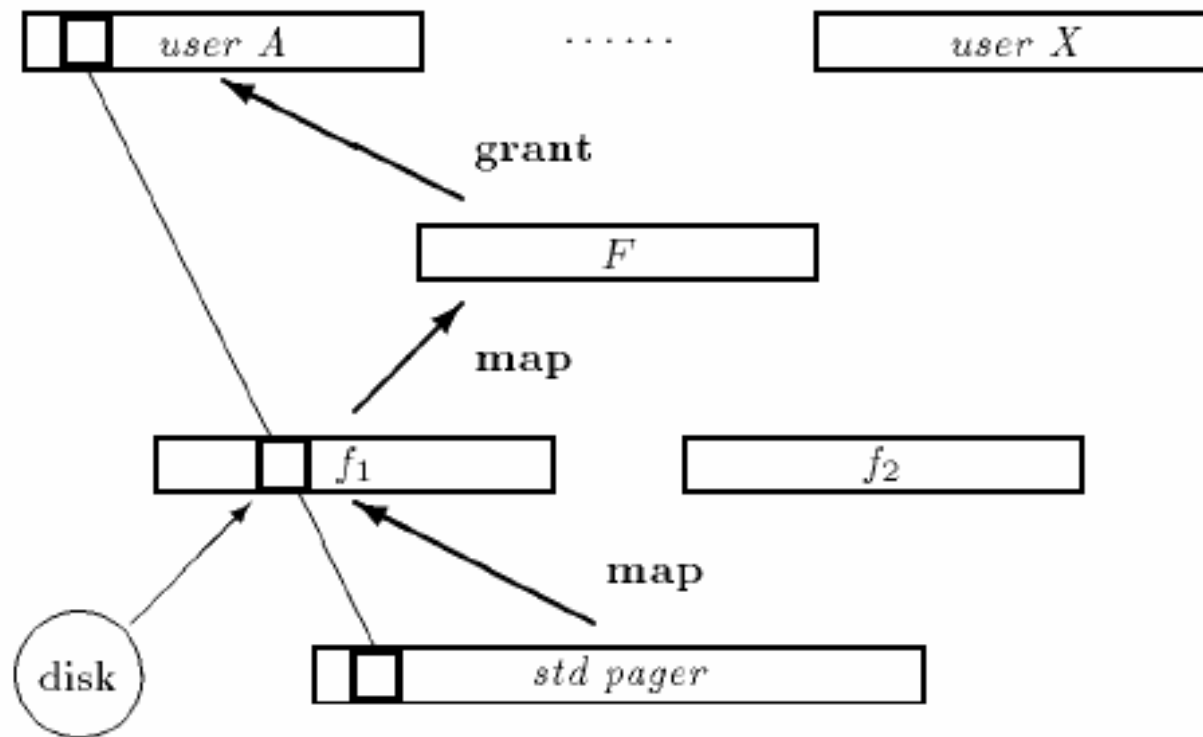
Espace d'adressage (1/2)

- Couche d'abstraction et de sécurité dépendante du matériel (memoire virtuelle paginée)
- Récursivité (sigma0) et opérations élémentaires (grant, map et flush)
- I/O ports = mapping





Espace d'adressage (2/2)





Threads et IPCs

- Thread: espace d'adressage + ensemble de registres avec au moins IP, SP et état
- Seul le kernel peut modifier un espace d'adressage
- Il doit donc gérer les IPC pour que les processus communiquent entre eux
- Utilisation des interruptions matérielles





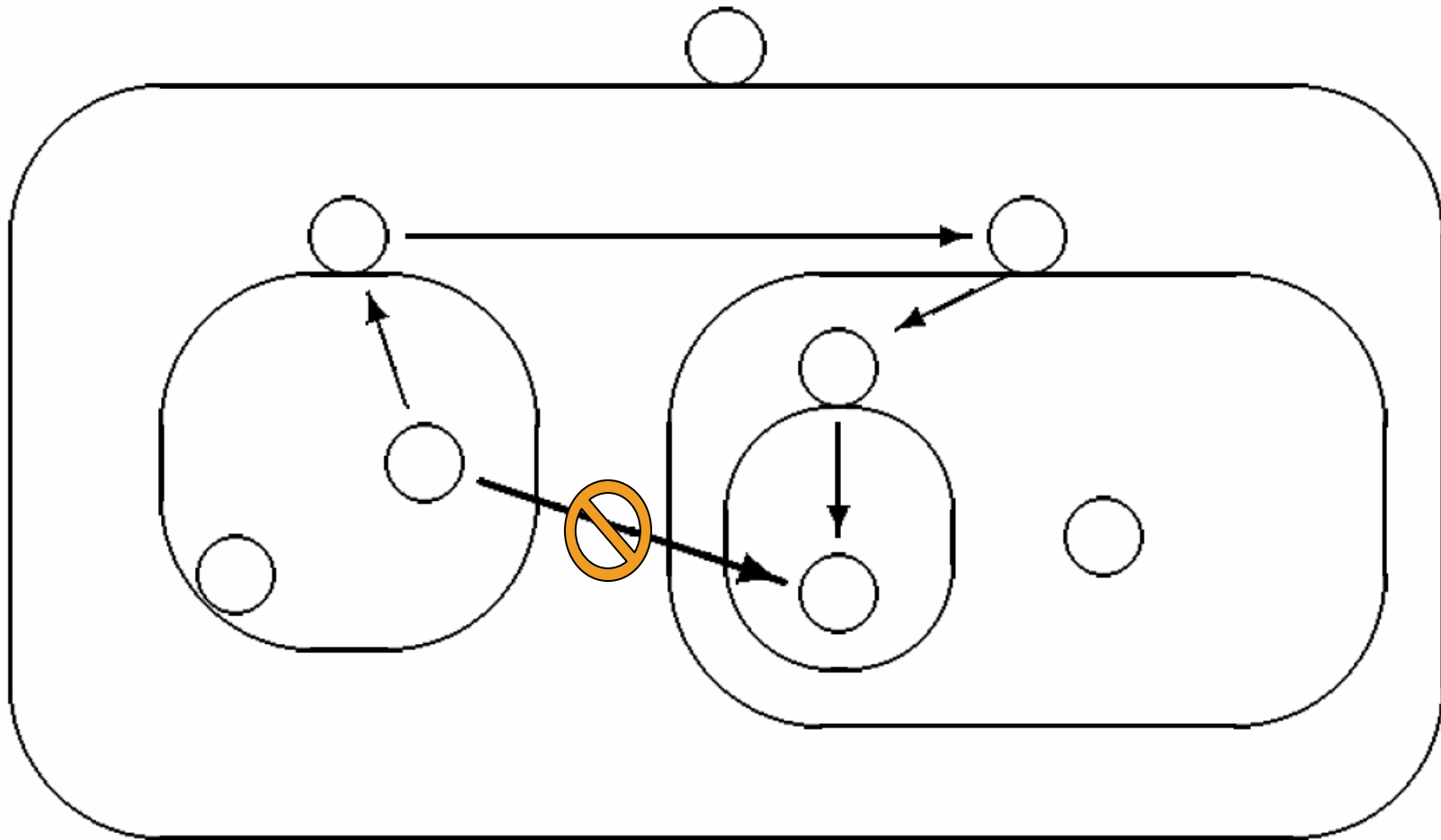
Clan and Chiefs (1/3)

- ACL niveau serveur
 - Dur à cause des changements de droits rapides
- Idée sur le principe d'Härlig et al [1993]
 - Le kernel filtre les messages sortant d'une tâche
- Un clan : ensemble de tâches
- Un clan, un chef
- Tout message sortant d'un clan passe par le chef
- Un chef est une tâche niveau utilisateur
 - Gestion fine et simplifiée





Clan and Chiefs (2/3)





Clan and Chiefs (3/3)

- Différentes organisations de clans typiques :
 - Clan par machine
 - Clan par version du système
 - Clan par utilisateur
 - Clan par tâche
- Niveau performance :
 - Intra-clan : négligeable
 - Inter-clan : dépend du nombre de chef mais
 - L3 IPC rapide
 - Communications inter-clan rares





Flexibilité

- Gestionnaire mémoire:
 - On peut gérer différents niveau d'espaces d'adressage.
- Pagination:
 - On peut aussi gérer différents paginateurs
- Allocation des ressources:
 - Contrainte pour le temps-réel
- Pilotes matériels:
 - Peuvent tourner en espace user.





Quelques Projets





FIASCO

- Implémente l'ABI de L4/x86
- Base de DROPS (Dresden Real Time OPerating System) à la fois temps réel et partagé
- Comble les lacunes de L4/x86 en temps réel (preemptibilité quasi permanente)
- C++ is good for you, Penguinized (GPL'ed)





L4::Pistachio



- Ecris en C++
- Disponible pour beaucoup d'archis
 - Linux 2.4 modifié disponible tournant au-dessus
- Première implémentation de l'API L4 version 4
 - Registres Virtuels
 - Threads et Espace d'adressage
 - IPCs super-rapides
 - Support multiprocesseurs
 - Interruptions
 - Threads privilégiés





Finissons en





Webographie

- L'évidence :
http://en.wikipedia.org/wiki/L4_microkernel_family
- Communauté de développeur de L4 :
<http://www.l4hq.org/>
- La famille de noyau L4 :
<http://os.inf.tu-dresden.de/L4/>
- Le projet L4Ka :
<http://l4ka.org/>
- Hommage a Jochen Liedtke :
<http://i30www.ira.uka.de/aboutus/people/liedtke/inmemoriam.php>





Conclusion

- Les micro noyaux remplaceront les noyaux monolithiques
 - Limite de maintenabilité
 - Fonctionnalités supplémentaires
 - Sécurité accrue





Questions ?

