Ecole Nationale des Sciences de l'Informatique, Pôle RIM

"Performance Evaluation of Commodity Systems for Cluster Computing"

DEA en Informatique

Imed Chihi <imed.chihi@ensi.rnu.tn>

Soutenu le 17 Juin 2003 devant le jury d'examen:

Président	**A. Belghith**
Membre	**L. Saidane**
Directeur	**F. Kamoun**
Co-encadrant	**A. Elleuch**

Sommaire

- Contexte
- Problématique
- Démarche
- Plates-formes cibles
- Quelques mesures
- Conclusion

Contexte et tendences

Calcul à hautes performances :

- Super calculateurs
- Grappes (“clusters”) d'ordinateurs
- Environnements d'execution et de développement distribués

Grappes de PCs

- Les stations de travail et les PC
  - Technologie de masse
  - En constante amélioration
- Un OS : Unix
  - OS largement utilisé et en constante amélioration
  - Non spécifiques et à usage général
- Environnements d'exécution et de développement
  - Au dessus de l'OS
  - Une machine virtuelle pour le calcul parallèle

Problématique

- Quels seraient les surcoûts introduits par les supports logiciels pour l'exploitation d'une grappe ?

- Est-ce que ces surcoûts varient énormément d'un système à un autre ?

Démarche

Niveaux d'abstraction :

- Application
- Supports pour le calcul parallèle
  - Outils de développement, librairies, serveurs, ordonnonceurs, etc.
- Système d'exploitation
  - Fichiers, sockets, processus, etc.
- Matériel
  - CPU, mémoire, DMA, etc.

- L'étude se limite à l'évaluation des surcoûts introduits sur **un** seul noeud de calcul par le **système d'exploitation**

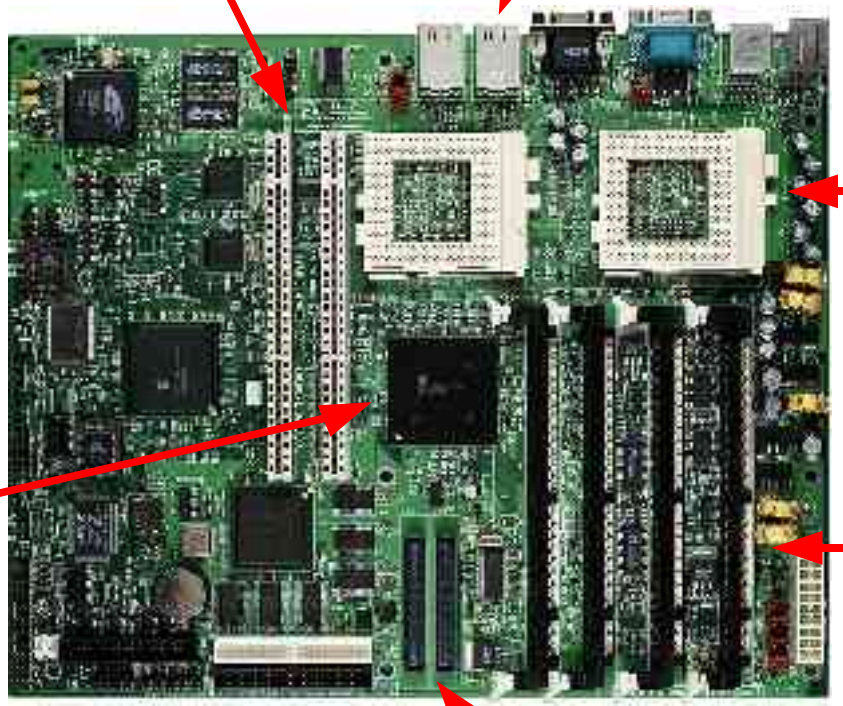
Démarche

- Décomposition des performances de l'OS
  - Commutation de contexte
  - Copie mémoire
  - Entrées/sorties
  - Création de processus
  - Signaux
  - ...
- Evaluation à travers des mesures : utilisation de micro-benchmarks (lmbench)

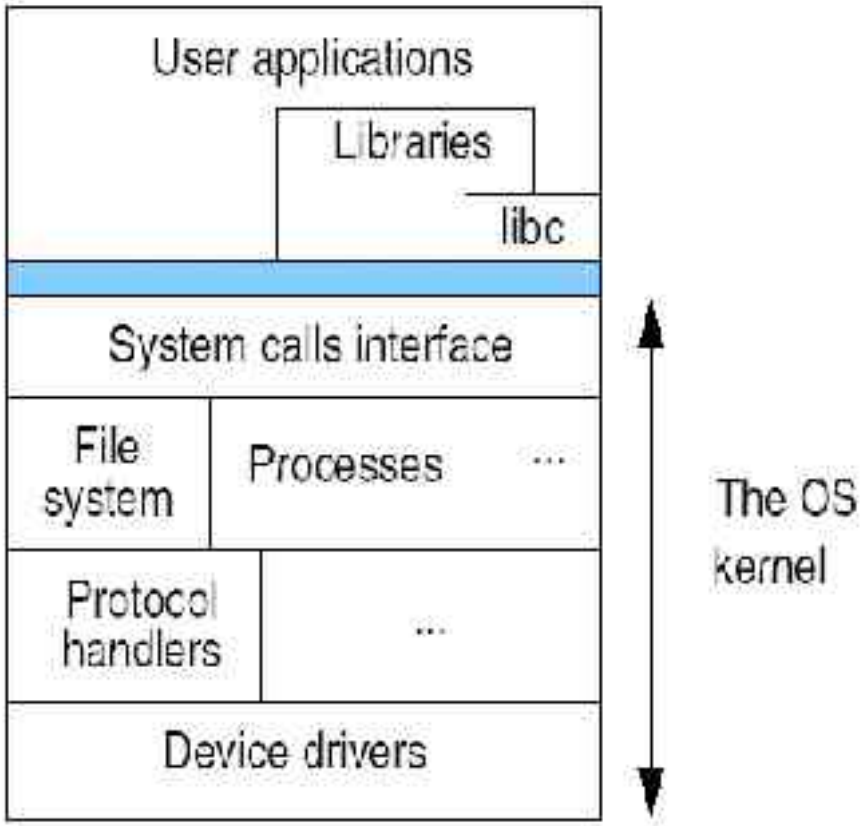
“Commodity systems”

- Matériel
  - Intel x86
  - Sun Sparc
  - Ethernet
- OS
  - Linux
  - Solaris

Un PC moderne



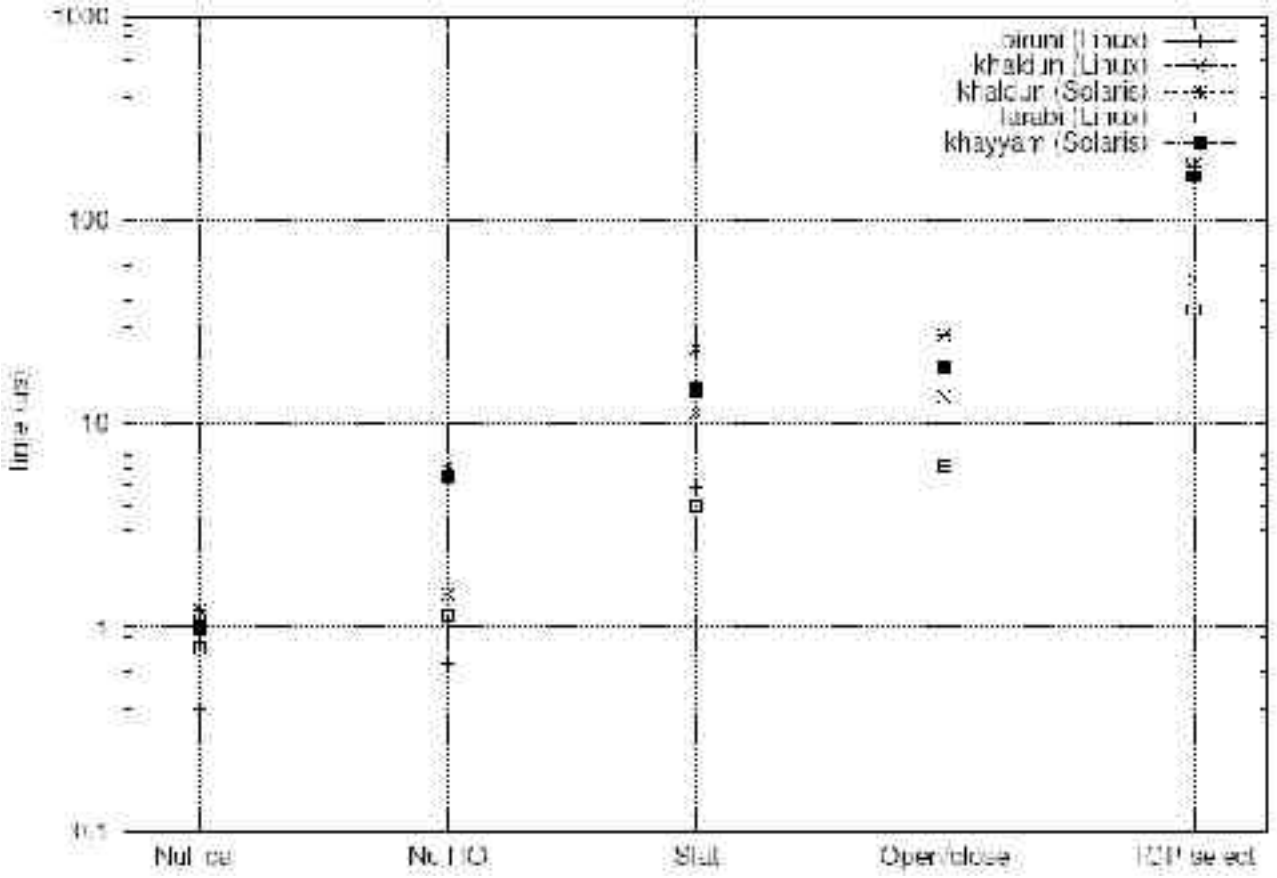
Le système d'exploitation



Mesures

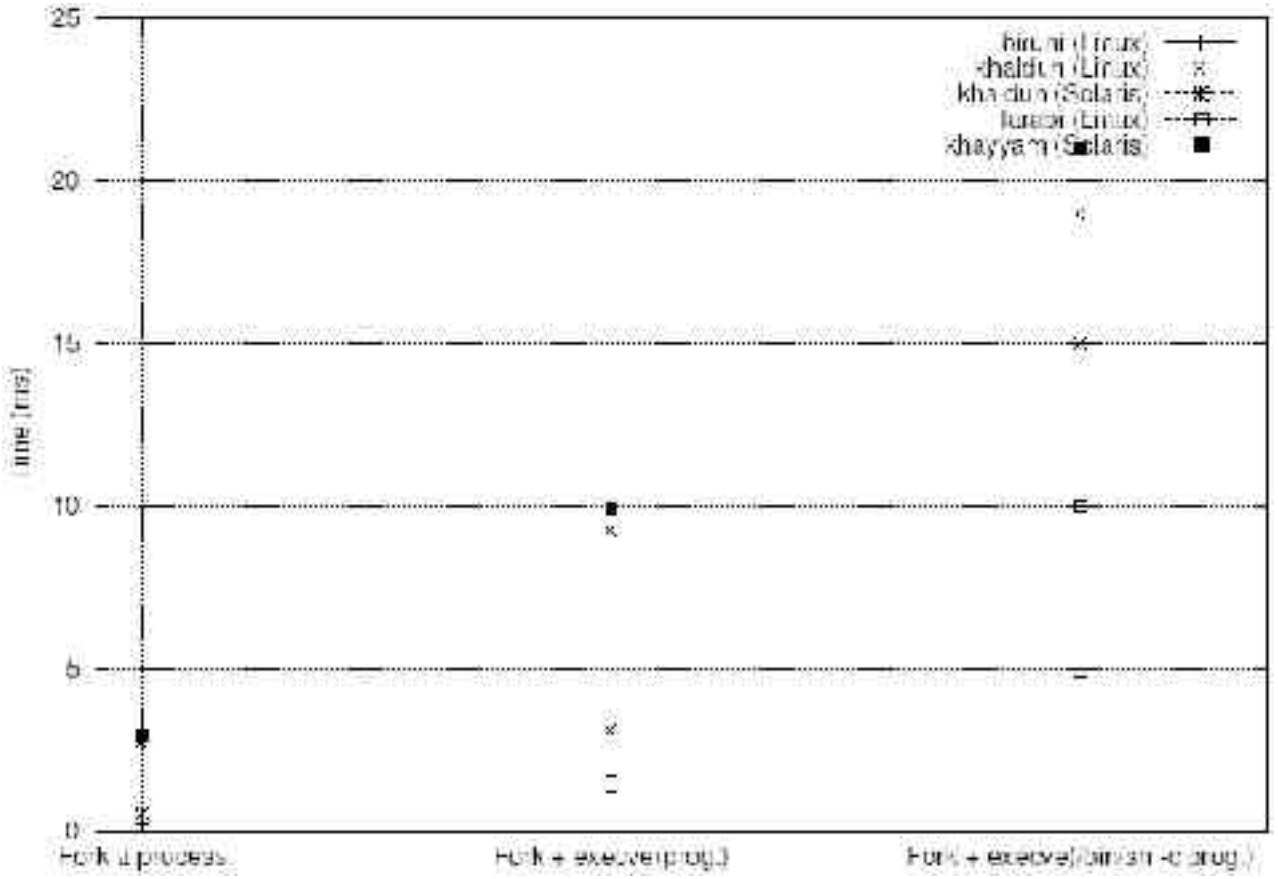
- Appels système
- Création de processus
- Commutation de contexte
- Communication Inter-Processus
- Copie mémoire
- Latence Ethernet vs. TCP
- Performances RAID (débit et usage CPU)
- Performance de mmap

Latence des appels système



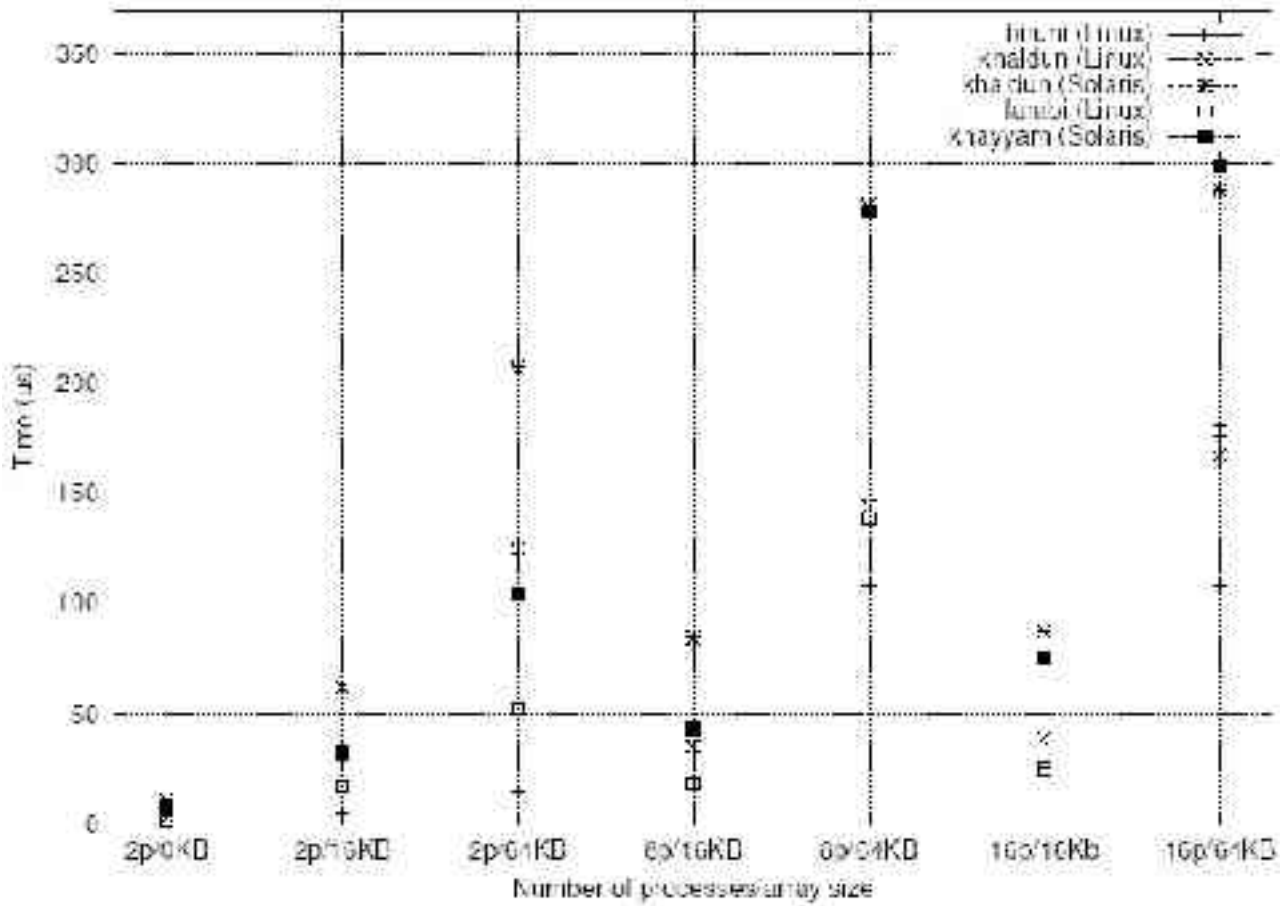
- Vitesse du CPU et les appels nuls
- Coût de select() en Mflops

Création de processus



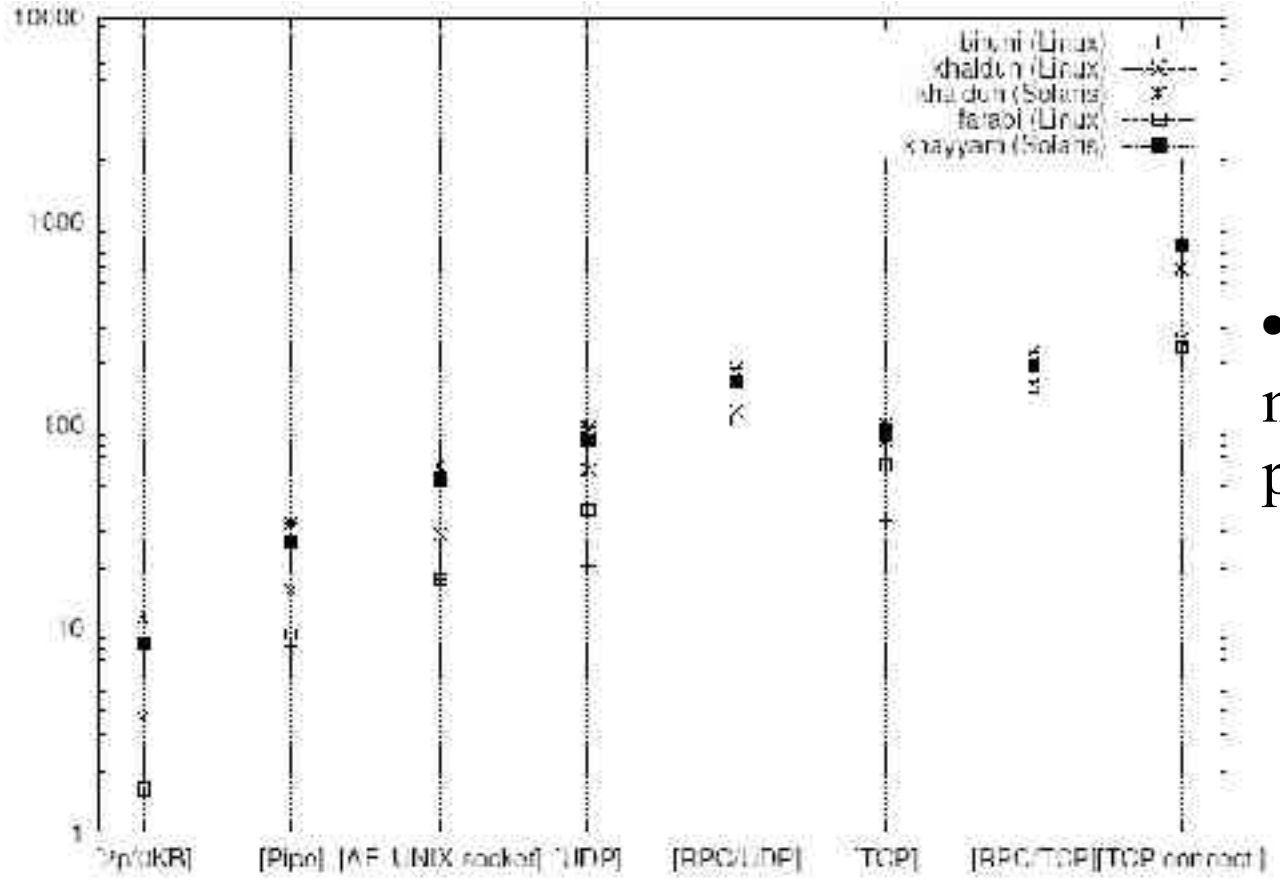
- La création de processus coûte cher!

Commutation de contexte



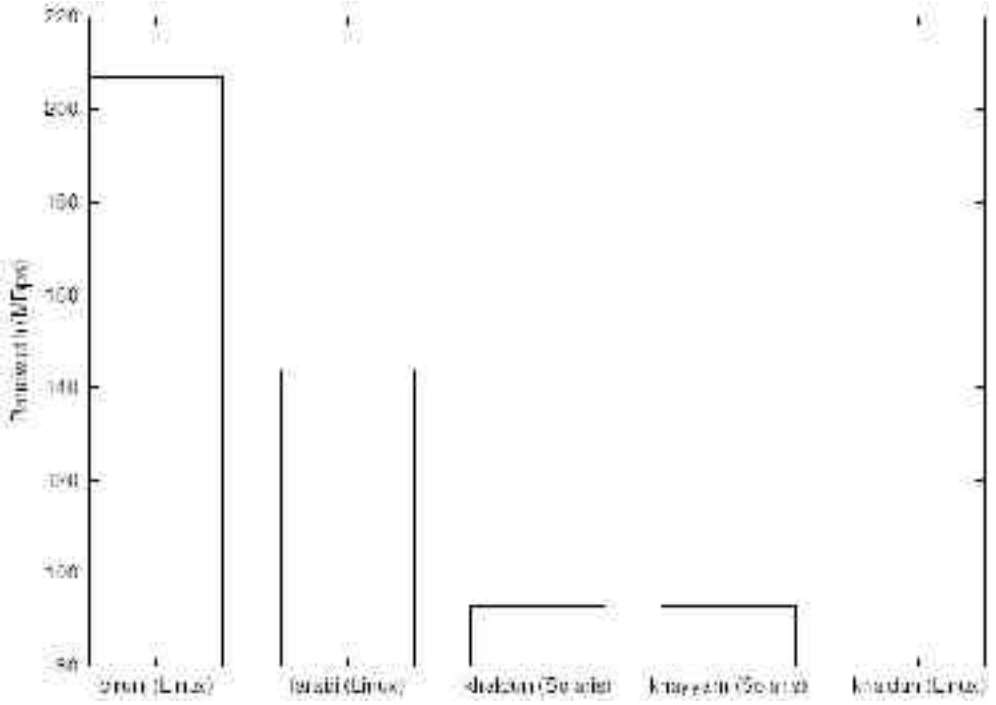
- Les données polluent les caches causant des commutations plus longues
- Le nombre de processus ne semle pas avoir un grand effet
- Eviter la concurrence sur un seul noeud

Communications

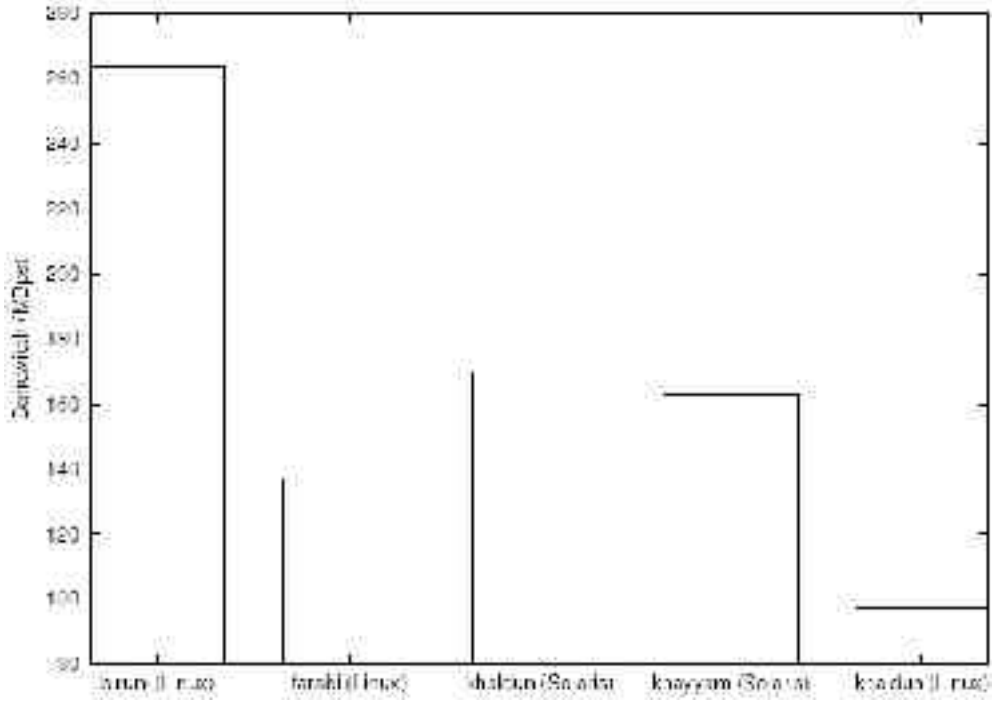


- Le coût croit avec les niveaux d'empilement de protocoles et leurs complexités

Débit de bcopy()



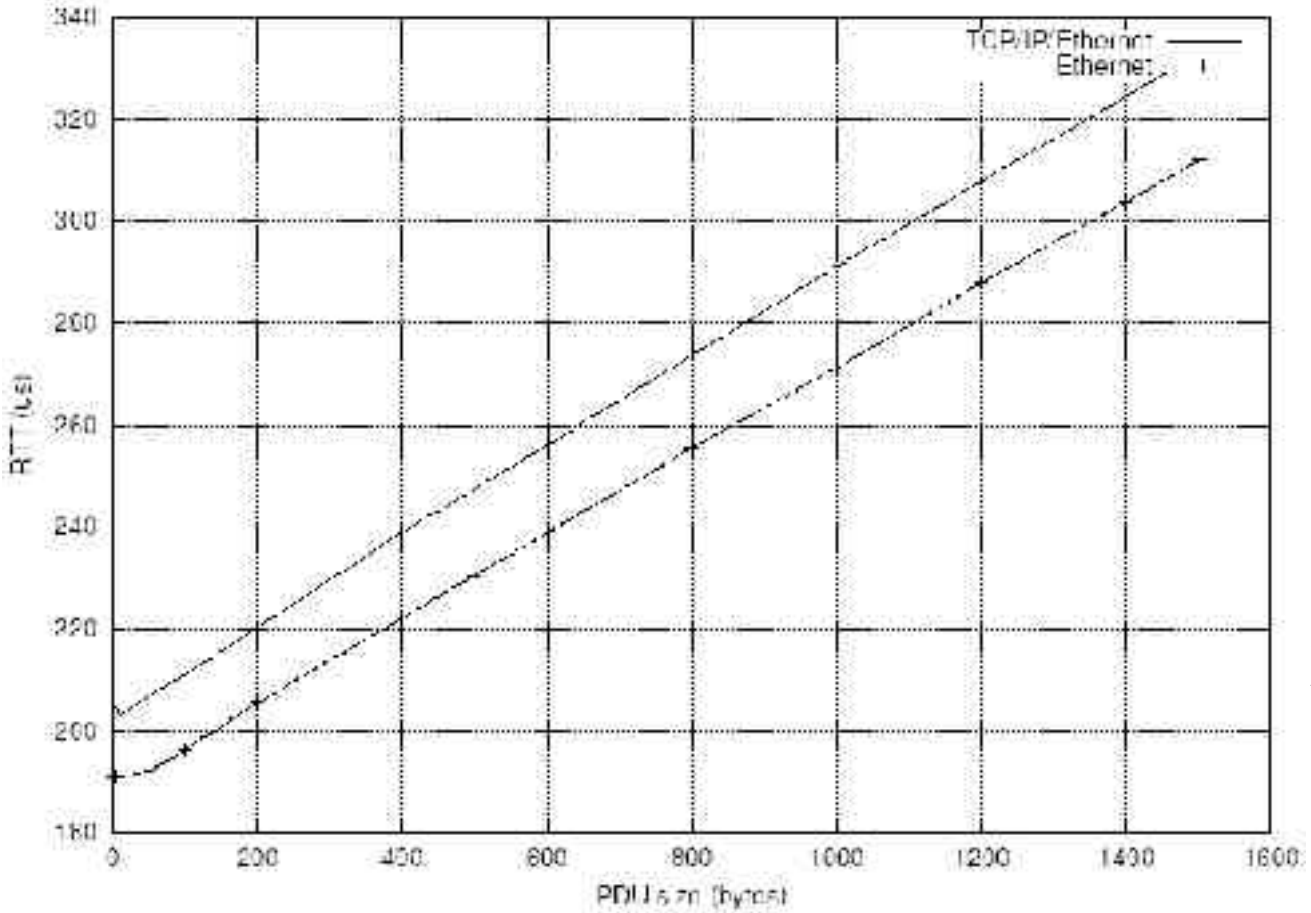
A la main



A travers libc

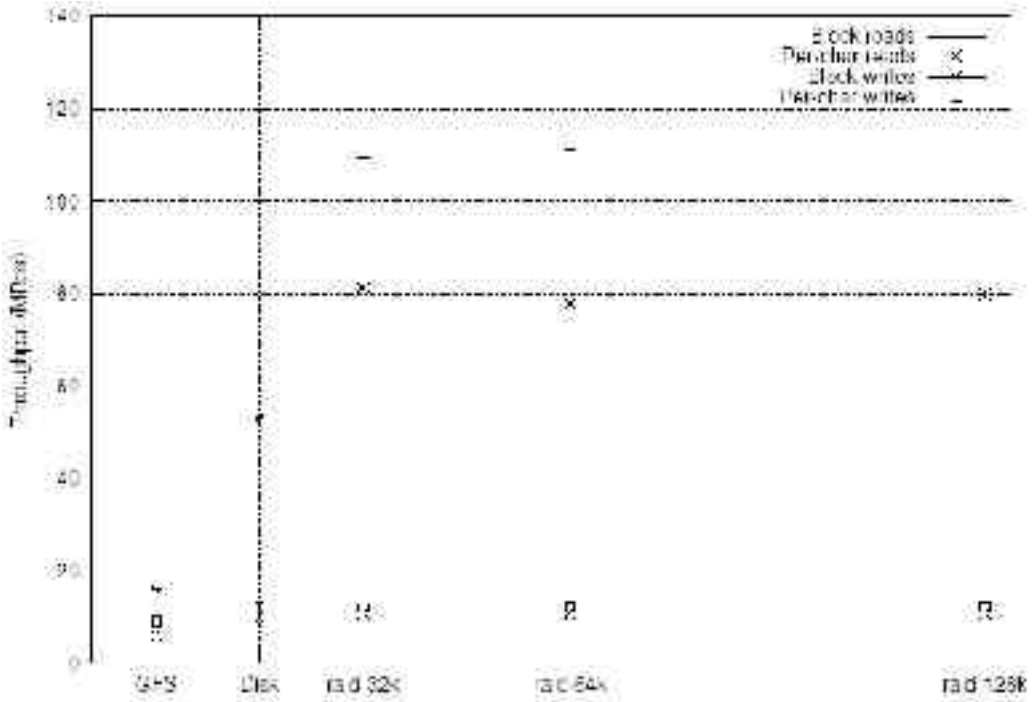
- Pas d'appels système
- A travers libc: plus rapide qu'à la main
- Copie à la main: pas de différence entre OSs

Latence de Ethernet vs. TCP

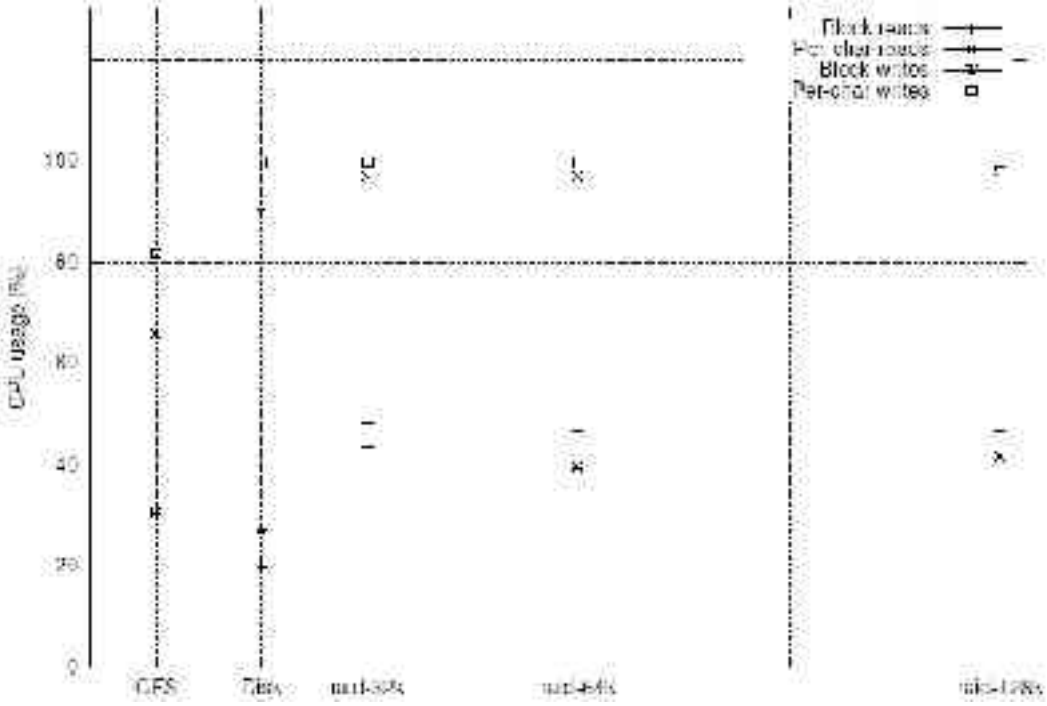


- Latence fixe plus importante que latence par octet
- TCP a un surcoût par octet peu significatif
- Le surcoût total de TCP est de 7% par rapport à Ethernet

RAID



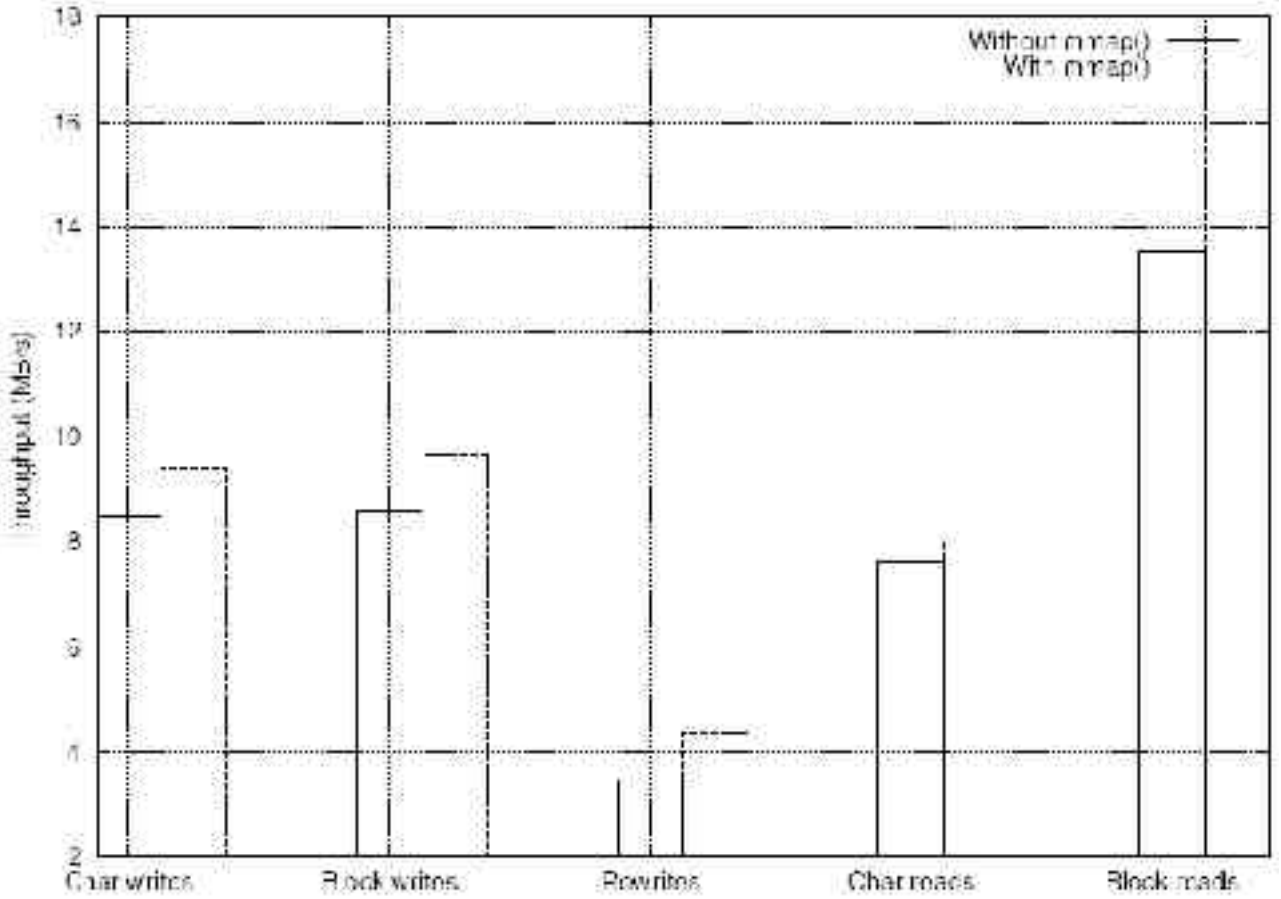
Débit



Utilisation du CPU

- Pas d'influence de la taille du bloc
- Le entrées/sorties par bloc sont plus rapides et moins couteuses
- Le CPU devient un goulot d'étranglement dans les opérations par caractère

Fichiers mappés



- Le mappage initial coûte
- Utile sur les fichiers ouverts une fois et accédés très souvent

Conclusions

- Linux est plus rapide que Solaris dans la plupart des tests
- Il est nettement plus efficace de dédier le processeur au calcul d'une seule tâche
- Les entrées/sorties par bloc sont à privilégier par l'usage de tampons circulaires (*ring buffers*)
- La traversée de la couche TCP dans le noyau n'introduit pas de surcoûts énormes