Brice Goglin
Équipe RESO - Laboratoire de l'Informatique du Parallélisme

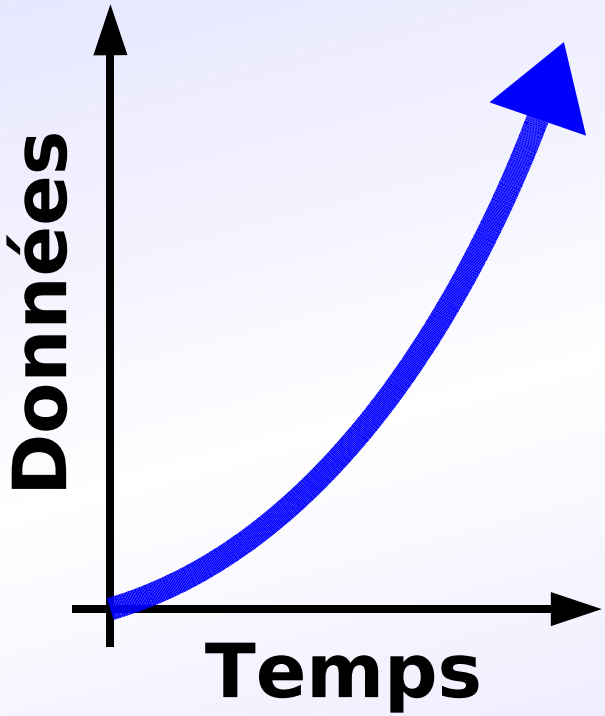
Réseaux rapides et stockage distribué dans les grappes de calculateurs :

propositions pour une interaction efficace

Thèse réalisée sous la direction de
Olivier Glück et **Pascale Vicat-Blanc Primet**

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
ECOLE NORMALE SUPERIEURE DE LYON
INRIA

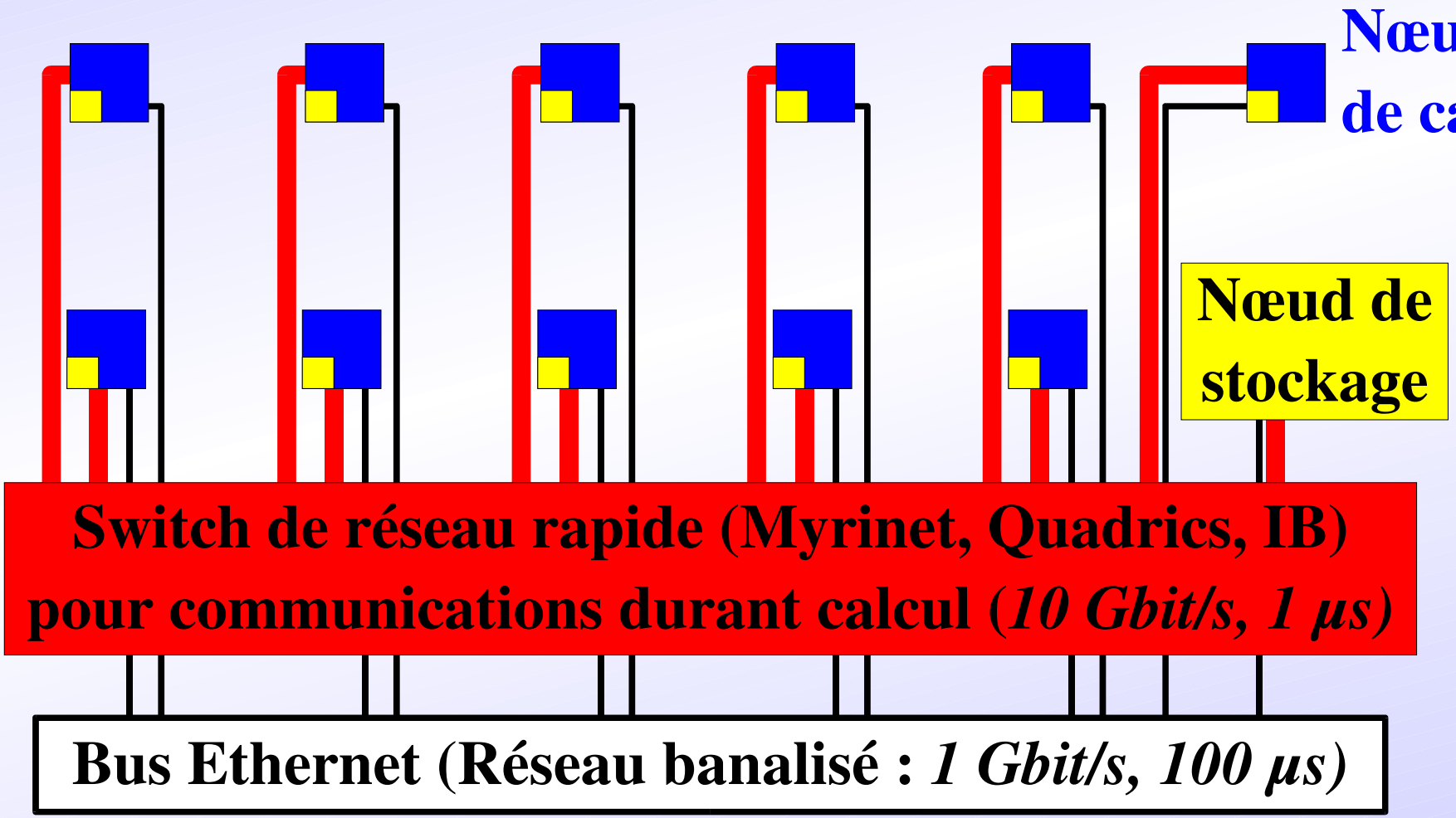
De plus en plus de données



15 Po par an au LHC

Diviser pour régner !

Anatomie des grappes de calcul



Problématique

- Stockage conçu pour les réseaux traditionnels
  - Forte latence (de 100 µs à 10 ms)

➔ Apport des réseaux rapides pour le stockage ?

- Réseaux rapides conçus pour applications MPI
  - Communications optimisées pour l'espace utilisateur

➔ Intégration des spécificités des réseaux rapides dans les systèmes de stockage distribué ?

  - Transferts de données
  - Contrôle des communications

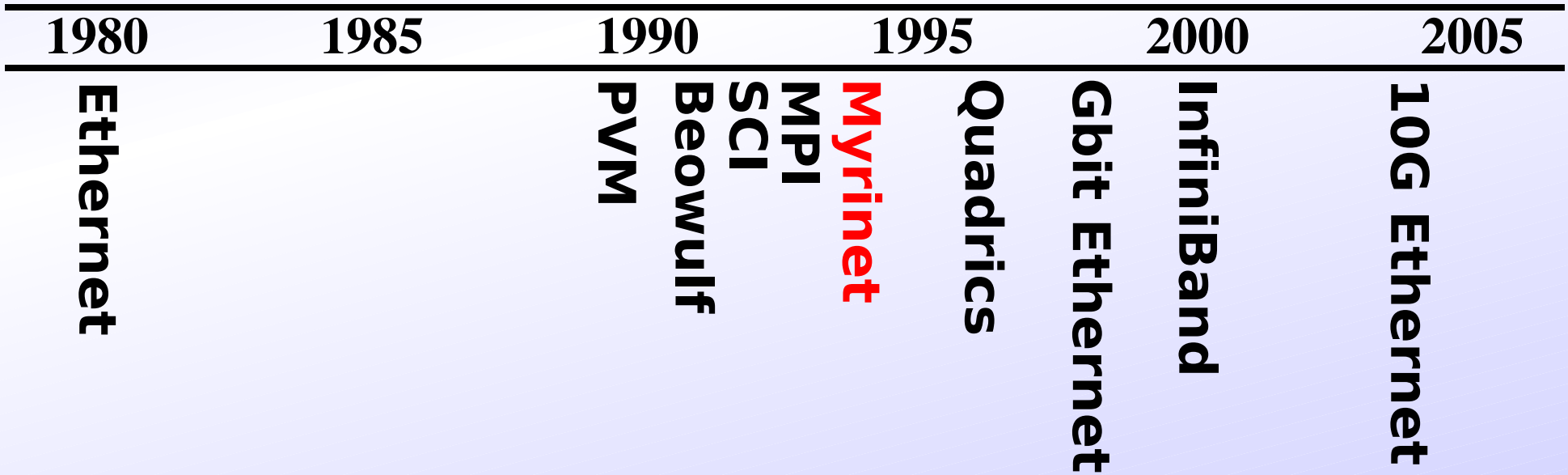
Contributions de la thèse

- Utilisation efficace des réseaux rapides pour le stockage distribué
  - Étude des transferts de données et du contrôle des communications
  - Mise en évidence des lacunes
    - dans les interfaces de programmation réseau
    - dans les systèmes d'exploitation
  - Proposition pour une interface de programmation réseau adaptée au stockage distribué
    - Intégration dans Myrinet Express

Plan de l'exposé

- Contexte
- Étude préliminaire
- Analyse des problèmes
- Propositions pour une interface noyau adaptée
- Conclusion et perspectives

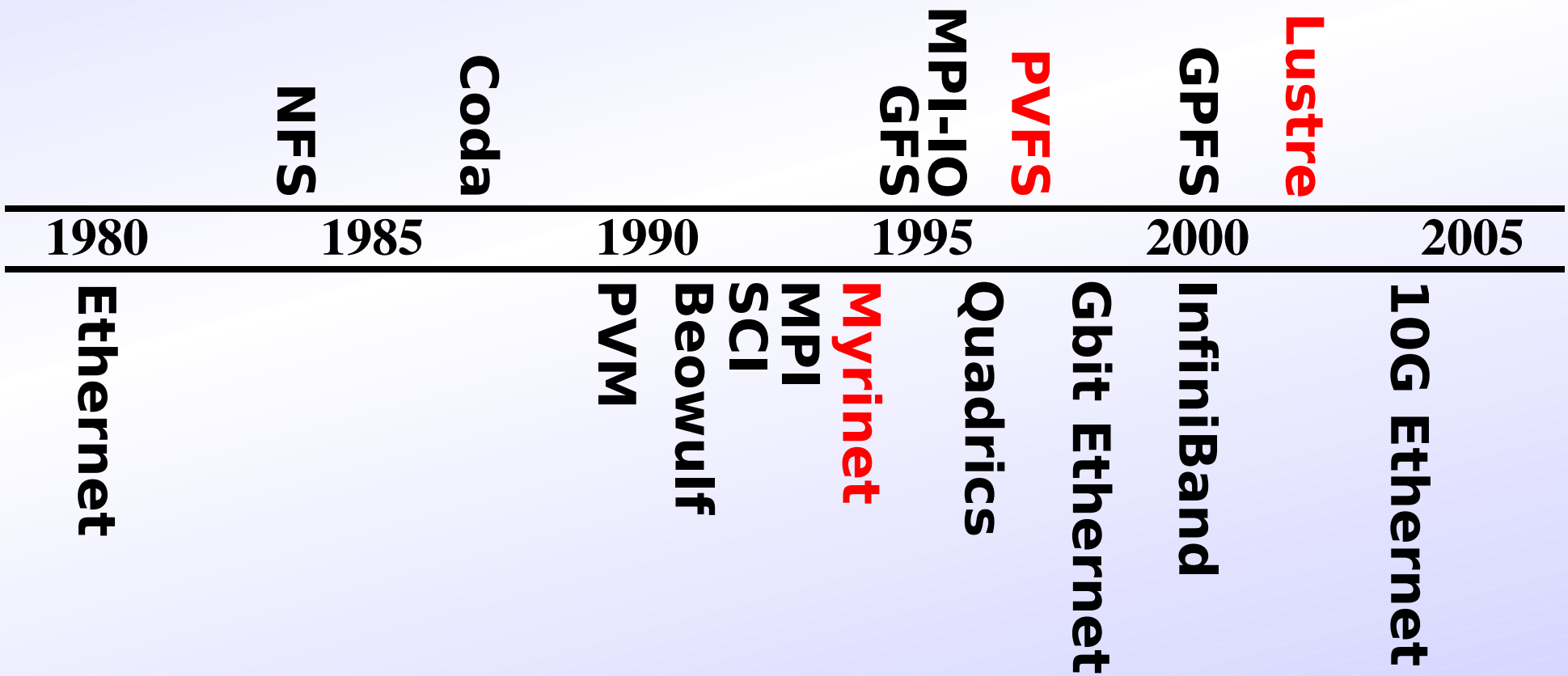
Évolution des réseaux



Des réseaux traditionnels aux réseaux rapides

Problèmes des réseaux traditionnels \ *Solutions dans les réseaux rapides*	Zéro-copie (DMA)	OS-Bypass	Réseau fiable	Traitement dans la carte	Interface asynchrone
Copies mémoire (~1 µs)	X	X			
Coût protocolaire (~10 µs)		X	X	X	
Absence de recouvrement				X	X

Évolution des systèmes de fichiers



Stockage performant dans les grappes

- Techniques de cache (NFS, Coda, ...)
  - Évite de solliciter le serveur
  - Très utilisé, même en dehors des grappes

[Sandberg,85]

- Parallélisation (PVFS, GFS, NFSp, Lustre, ...)
  - Permet d'augmenter la charge supportée par le serveur

[Ligon,99]

- Transferts performants (DAFS)
  - Utilisation du réseau rapide

[Magoutis,02]

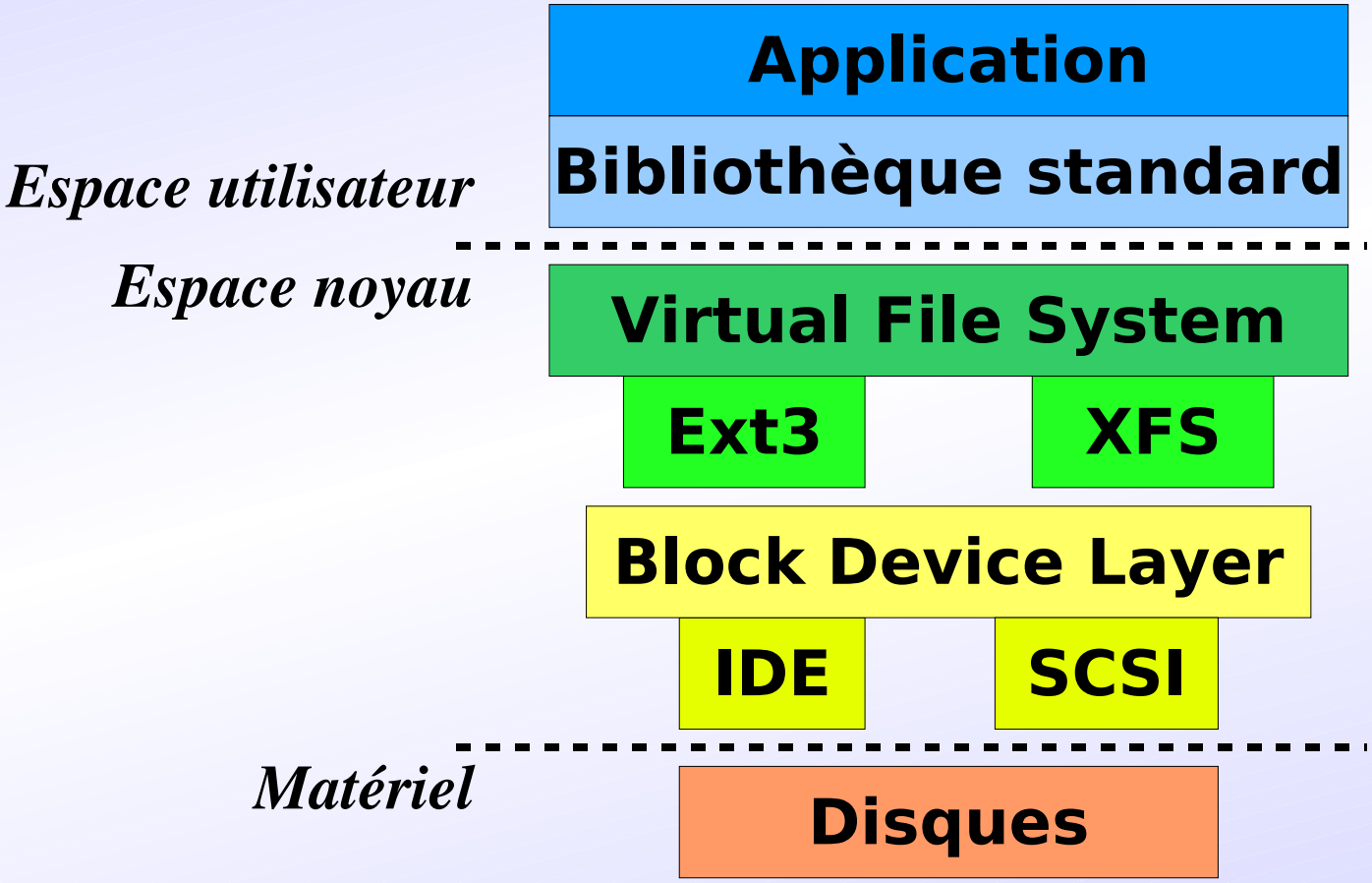
Utilisation des réseaux rapides pour le stockage distribué

- Apport de Myrinet sur PVFS indéniable [Carns,00]
  - Mais inférieur au gain attendu
- PVFS2 retourne en espace utilisateur pour les accès au réseau depuis le noyau [Ligon,01]

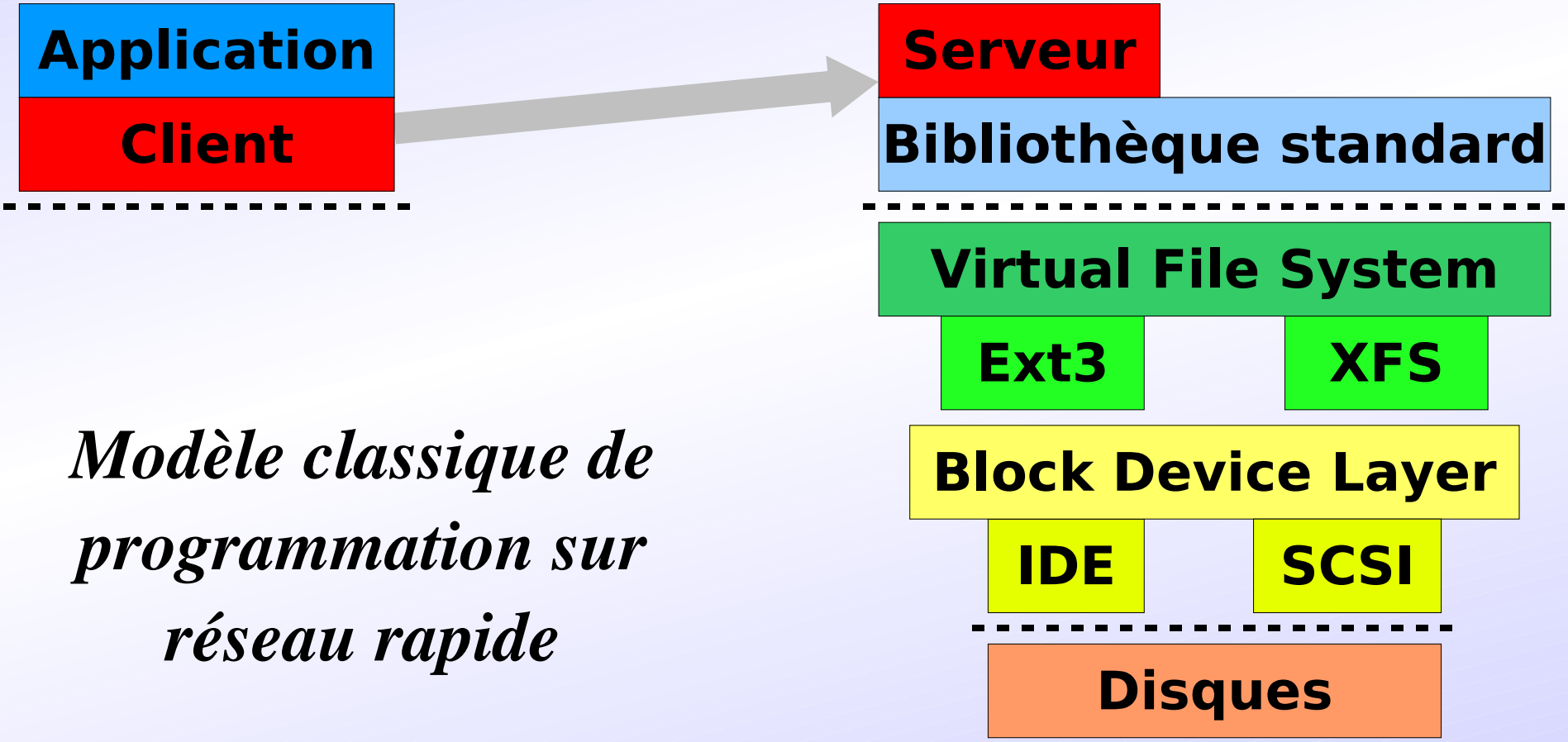
> "*it is not clear that we will have ready access to all networking APIs from within the kernel*"

- Lustre indisponible ou peu performant sur certains réseaux rapides
  - Utilise des copies mémoire sur Myrinet/GM

Accès au stockage local

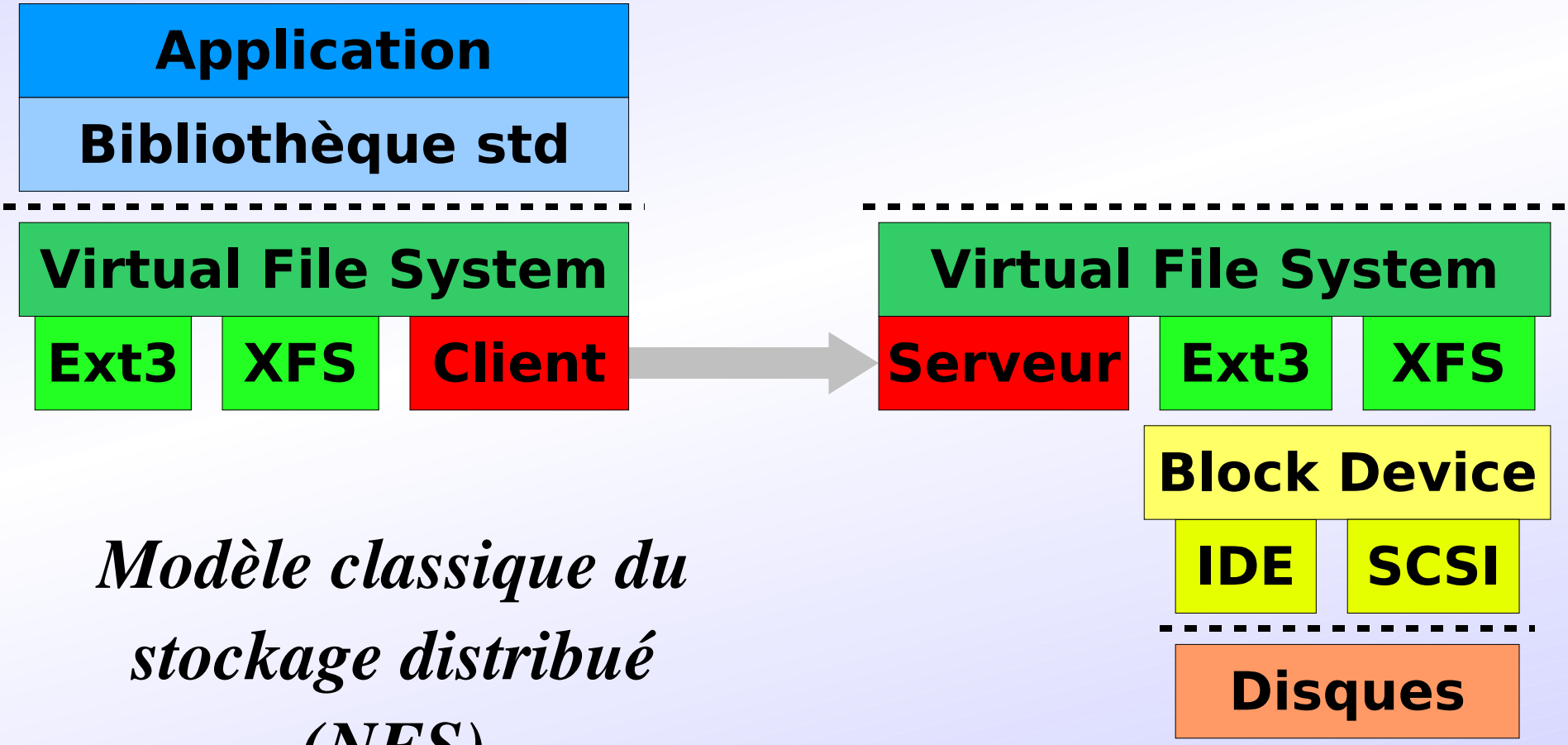


Accès au stockage distant en espace utilisateur



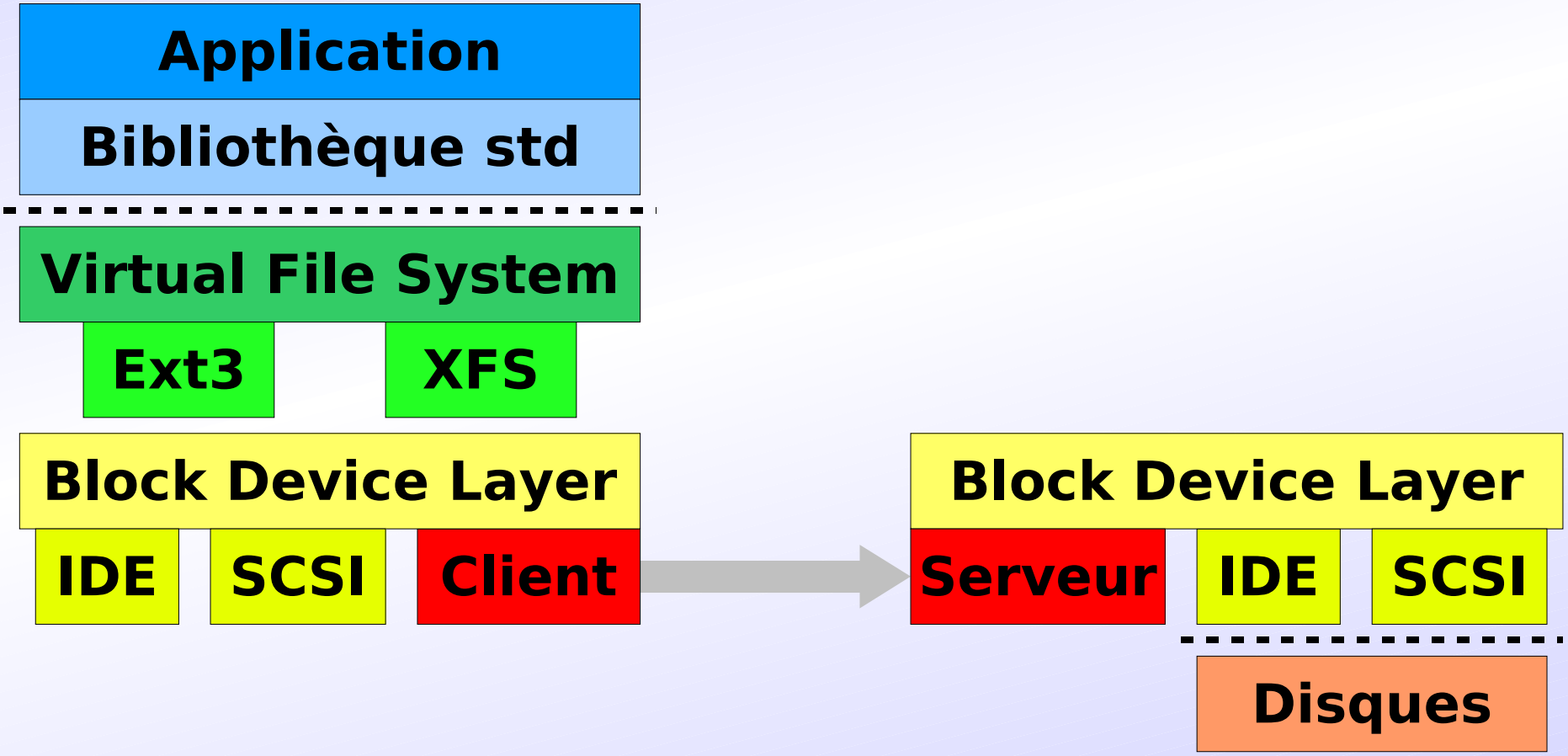
Modèle classique de programmation sur réseau rapide

Accès au stockage distant en espace noyau

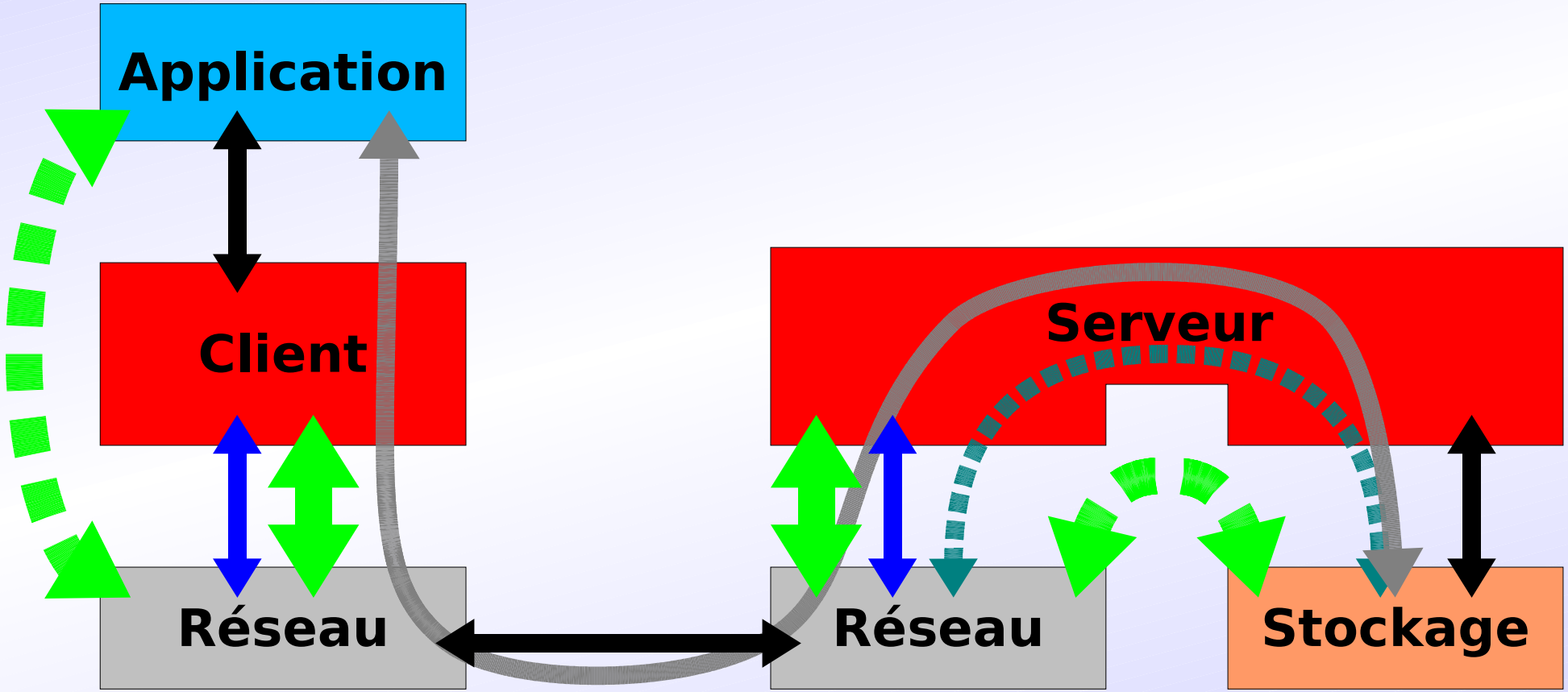


Modèle classique du stockage distribué (NFS)

Accès au stockage distant au niveau bloc



Analyse des interactions mises en jeu



$$T_{StoC} + T_{RegC} + T_{AsyncC} + T_{Reseau} + T_{RegS} + T_{AsyncS} + T_{EventS} + T_{StoS}$$

Plan de l'exposé

- Contexte
- Étude préliminaire
- Analyse des problèmes
- Propositions pour une interface noyau adaptée
- Conclusion et perspectives

Hypothèses et contraintes

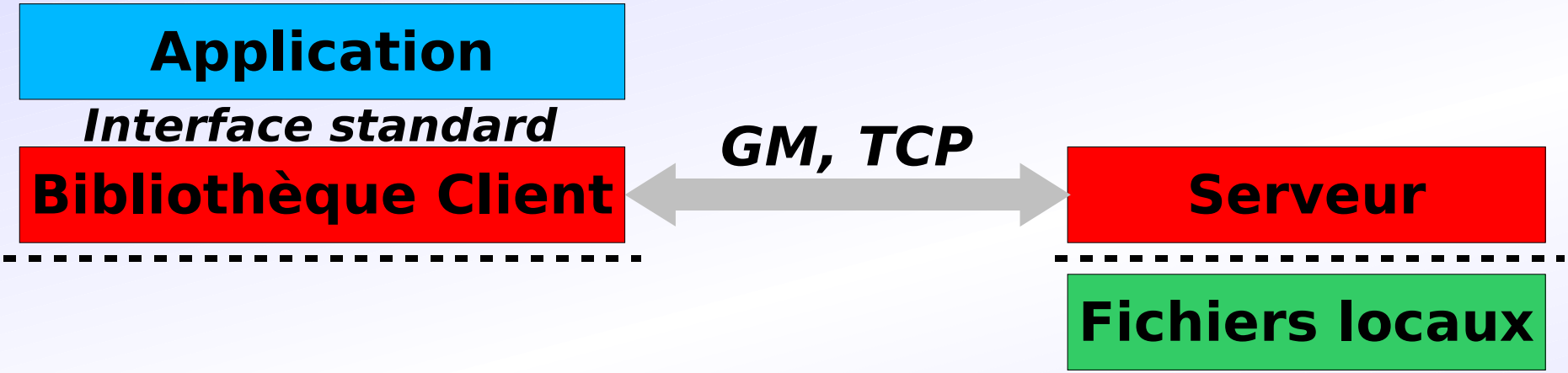
- Respect de l'interface standard de programmation
  - Accès standard aux fichiers distants depuis n'importe quelle application
- Modèle client-serveur (PVFS, Lustre)
  - Solution complémentaire aux stratégies classiques (cache et parallélisation)
- Réseaux Myrinet, pilote GM
  - Validation expérimentale

Démarche expérimentale

- Plate-forme expérimentale maîtrisée
  - Éliminer les bruits et analyser les différents coûts
- Étude préliminaire dans le cadre du modèle classique des réseaux rapides
  - Localiser les optimisations à proposer
- Comparer les différentes méthodes d'accès aux données distantes
  - Espace utilisateur, noyau, blocs

Étude préliminaire en espace utilisateur :

Expérimentations



- ORFA, *Optimized Remote File Access*
  - Chemin le plus rapide entre application et stockage
- Mesure des temps d'accès aux fichiers
  - Lecture/écriture par appels à read/write

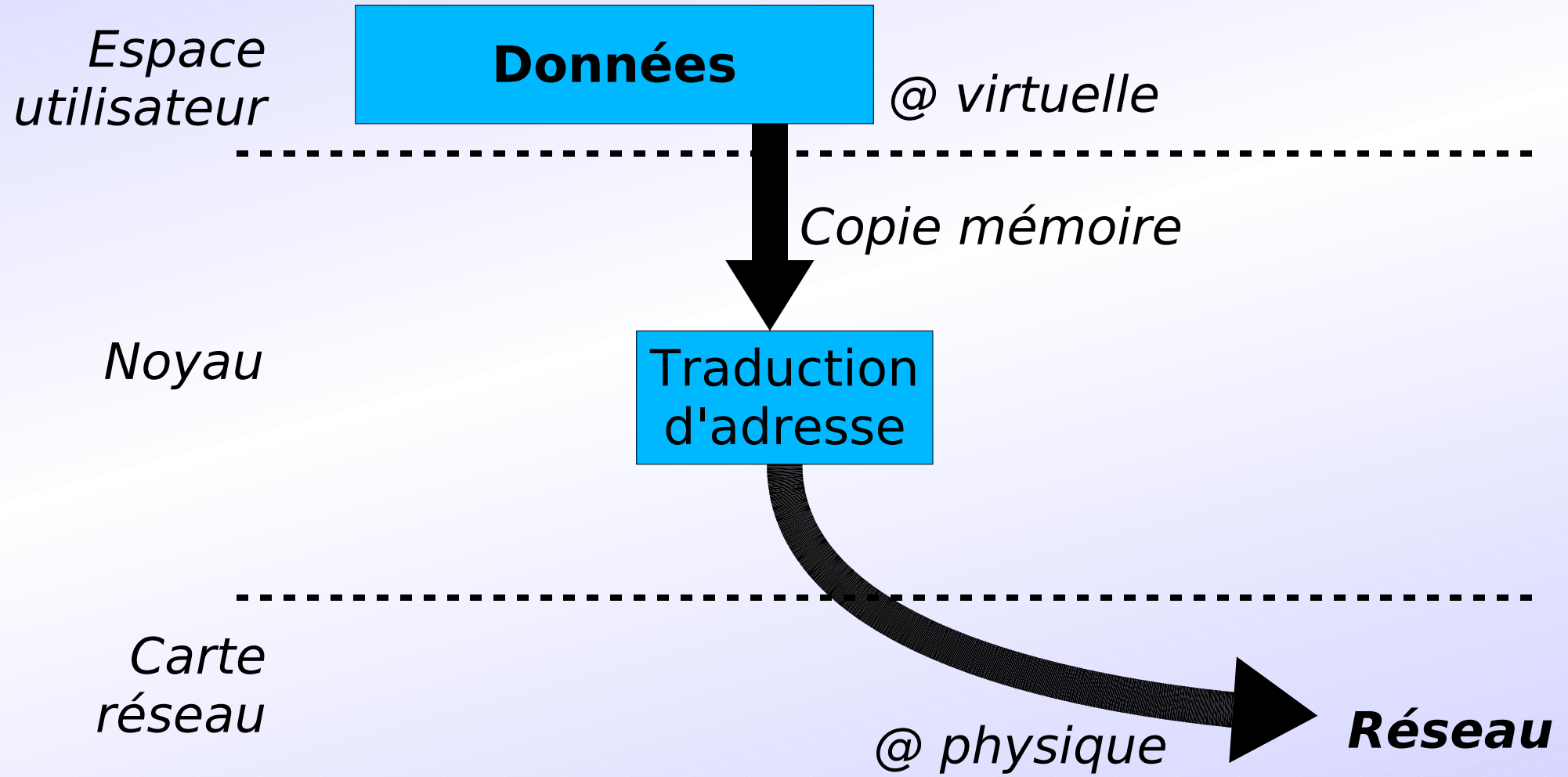
Étude préliminaire en espace utilisateur : Résultats expérimentaux

- Mesures avec 1 client et 1 serveur
  - Importance du débit du réseau
    - Gérer efficacement les transferts de données
- Avec ou sans cache
  - Faible impact de la latence
    - Ne pas écarter les stratégies basées sur un cache côté client
- Nombreuses écritures concurrentes
  - Dégradation des performances du serveur

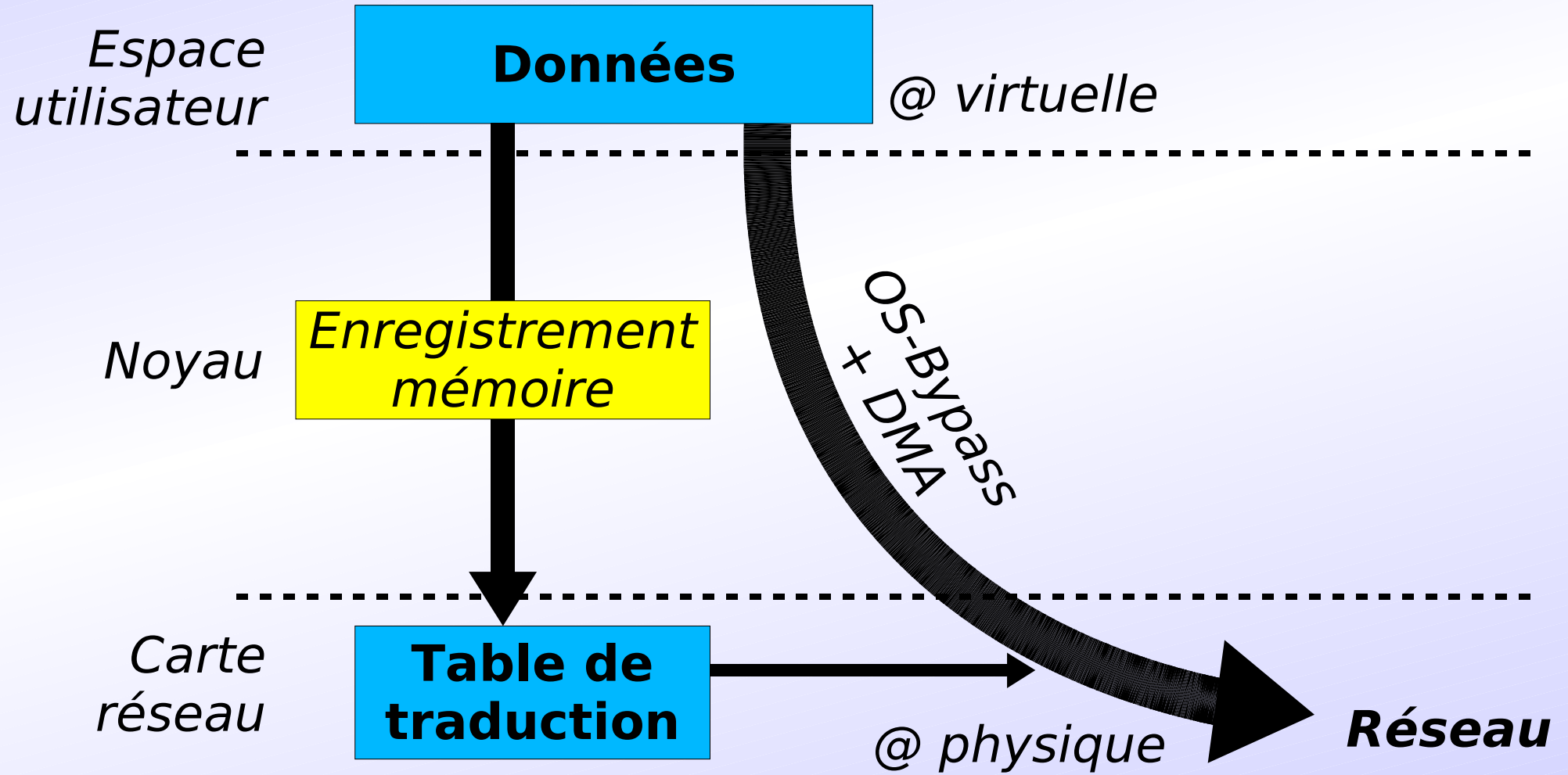
Plan de l'exposé

- Contexte
- Étude préliminaire
- Analyse des problèmes
- Propositions pour une interface noyau adaptée
- Conclusion et perspectives

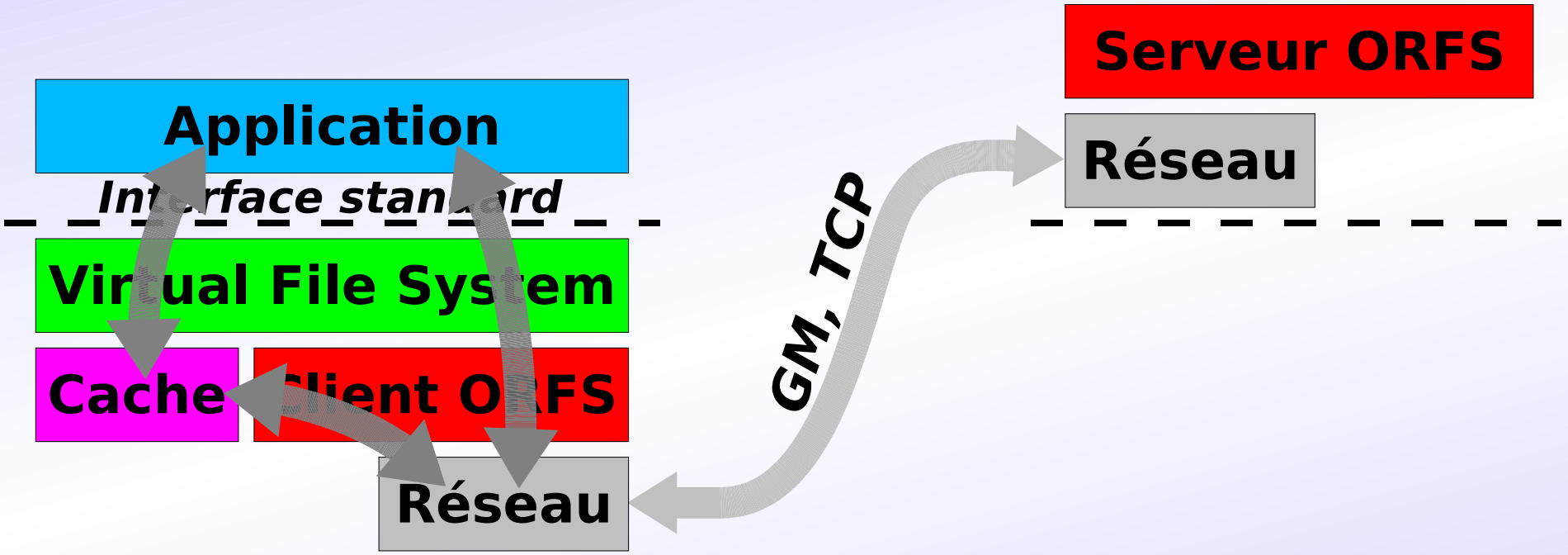
Communications traditionnelles



Communications Zéro-copie



Transferts de données dans le noyau



- Type d'accès choisi par l'application
  - Accès zéro-copie (`O_DIRECT`)
  - Accès par le cache du système d'exploitation

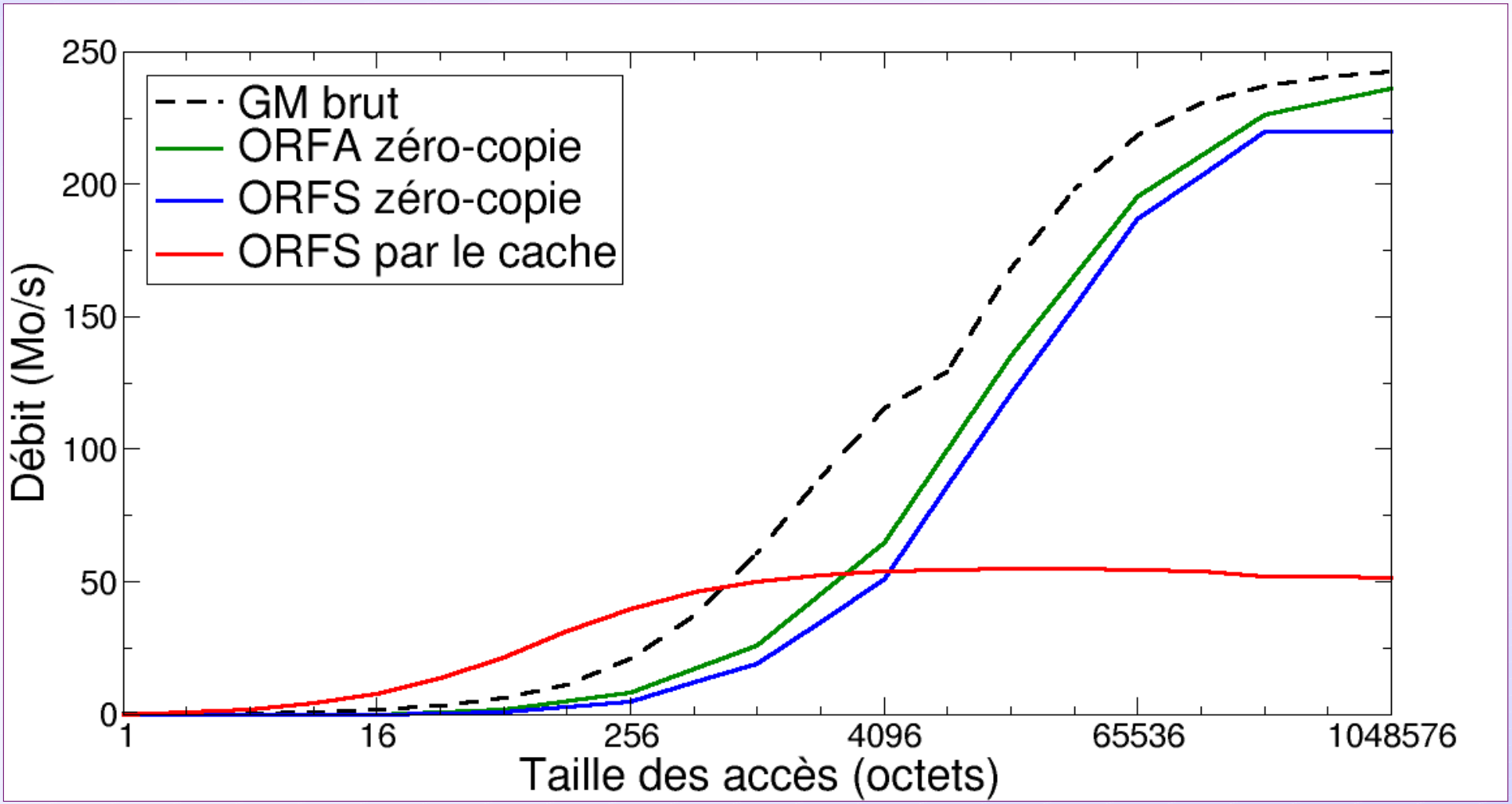
Accès zéro-copie depuis le noyau

- Enregistrement de pages utilisateur depuis le noyau
  - Distinguer les mêmes adresses virtuelles d'espaces d'adressage différents
    - ➔ Modification de l'interface GM et du MCP
- Maintenir le cache d'enregistrement à jour
  - Modification du système d'exploitation pour signaler les zones enregistrées invalides
    - ➔ Infrastructure VMA Spy dans le noyau Linux
- Performances proches de celles en espace utilisateur

Accès par le cache du noyau

- Enregistrement de pages du noyau ?
  - Pages verrouillées, non mappées en mémoire virtuelle, adresse physique disponible
    - → Enregistrement inapproprié
- Passage des adresses physiques à la carte réseau
  - → Ajout d'une interface GM basée sur l'adresse physique
    - → Modification du MCP
- Très simple à utiliser
- Gain en latence (1 µs)

Évaluation des performances



Notifications d'événements

- Le client attend
  - Terminaison d'une communication **particulière**
- Le serveur attend
  - Terminaison d'une communication **quelconque**
- Les deux stratégies rarement proposées à la fois
  - Émulation par attente active ou threads
    - Consommation processeur et latence (quelques µs)
- Besoins similaires à MPI

Trafic concentré

- Modèle client-serveur impose des messages inattendus
  - Risque de gaspillage du débit utile
- ➔ Les clients doivent attendre l'accord du serveur
  - Protocole contrôlé par le serveur
  - Accès mémoire à distance initiés par le serveur
    - Charge plus importante ?
- ➔ Nécessité d'un contrôle de flux efficace dans l'interface de programmation réseau
  - Même besoin que MPI

Plan de l'exposé

- Contexte
- Étude préliminaire
- Analyse des problèmes
- Propositions pour une interface noyau adaptée
- Conclusion et perspectives

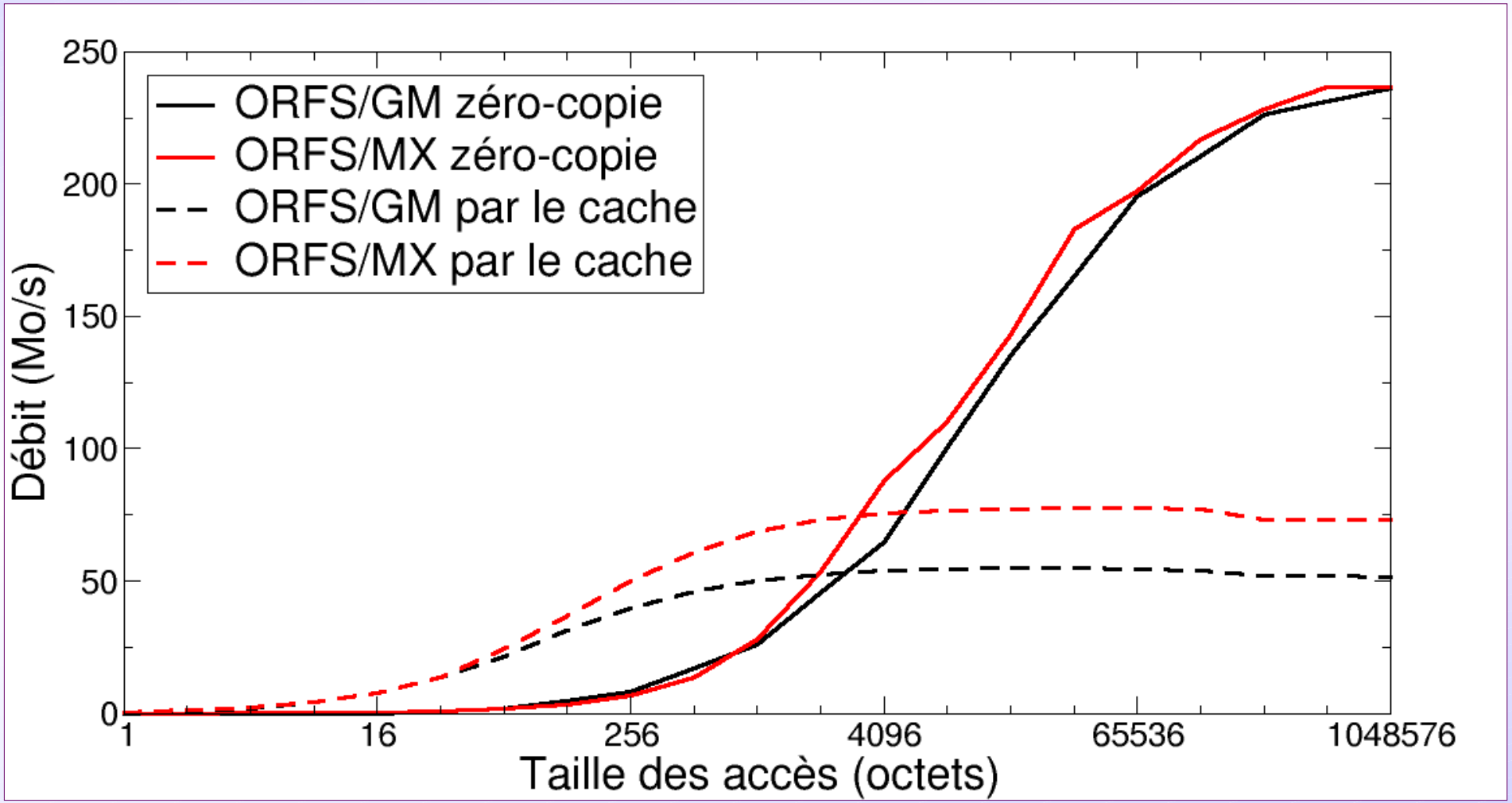
Le pilote Myrinet Express

- Matériel réseau Myrinet apparu en 1995
- Pilote officiel GM limité
- Nombreux travaux académiques (BIP, ...)
- Nouveau pilote officiel Myrinet Express
  - Interface de MX calquée sur MPI
    - Répond aux besoins en contrôle des communications
  - Intègre fonctionnalités avancées
    - Performances meilleures que GM

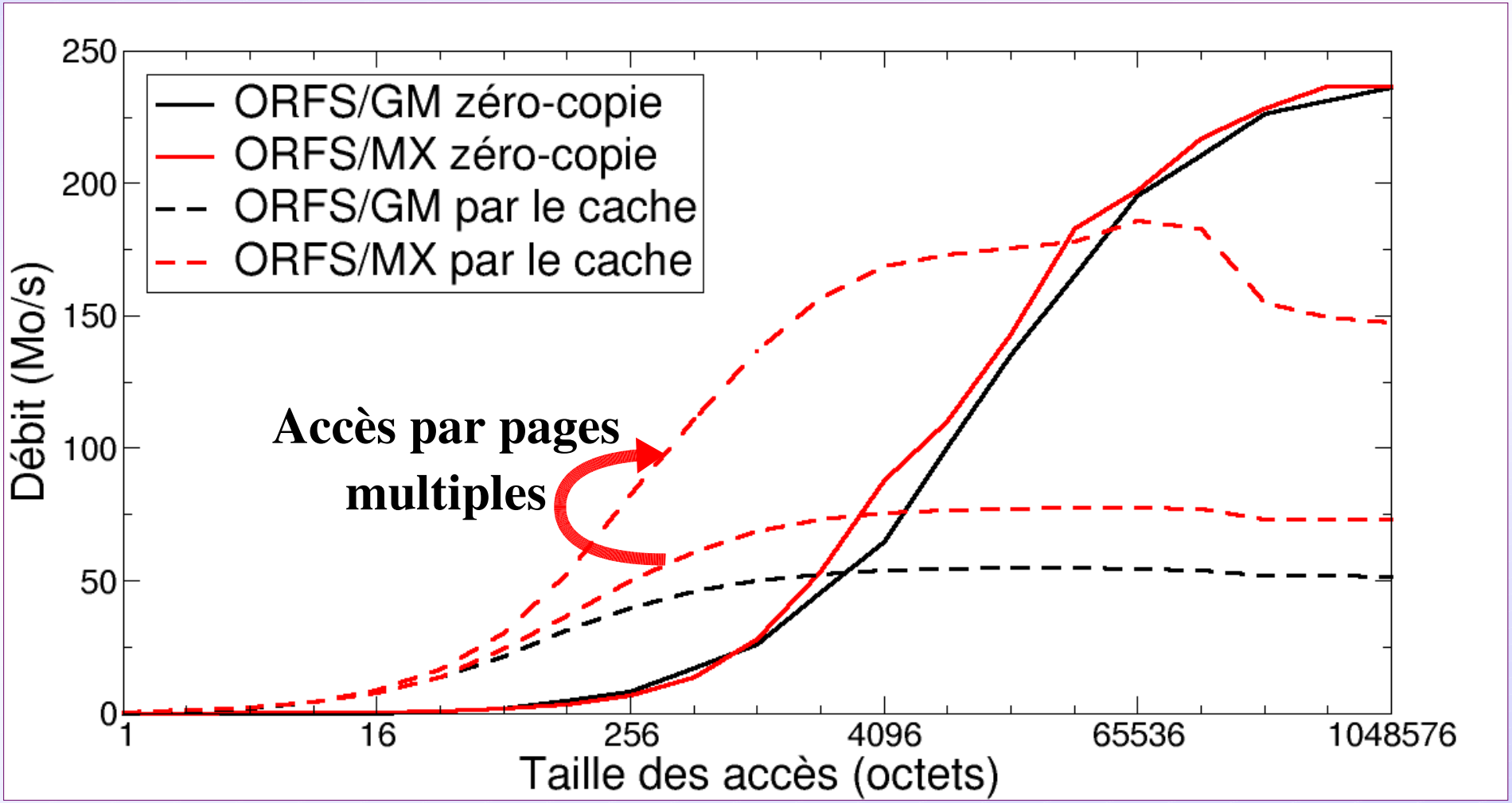
Apport :
Une interface noyau dans MX

- Exposition de l'interface MX dans le noyau
  - Optimisation pour ce contexte
- Extension gérant le type de mémoire mise en jeu
  - Adressage virtuel utilisateur, virtuel noyau ou physique
  - Assistance de l'application
    - Facile à utiliser
- Optimisations de la mise en œuvre en tenant compte du type de mémoire
  - Suppression de copies mémoire

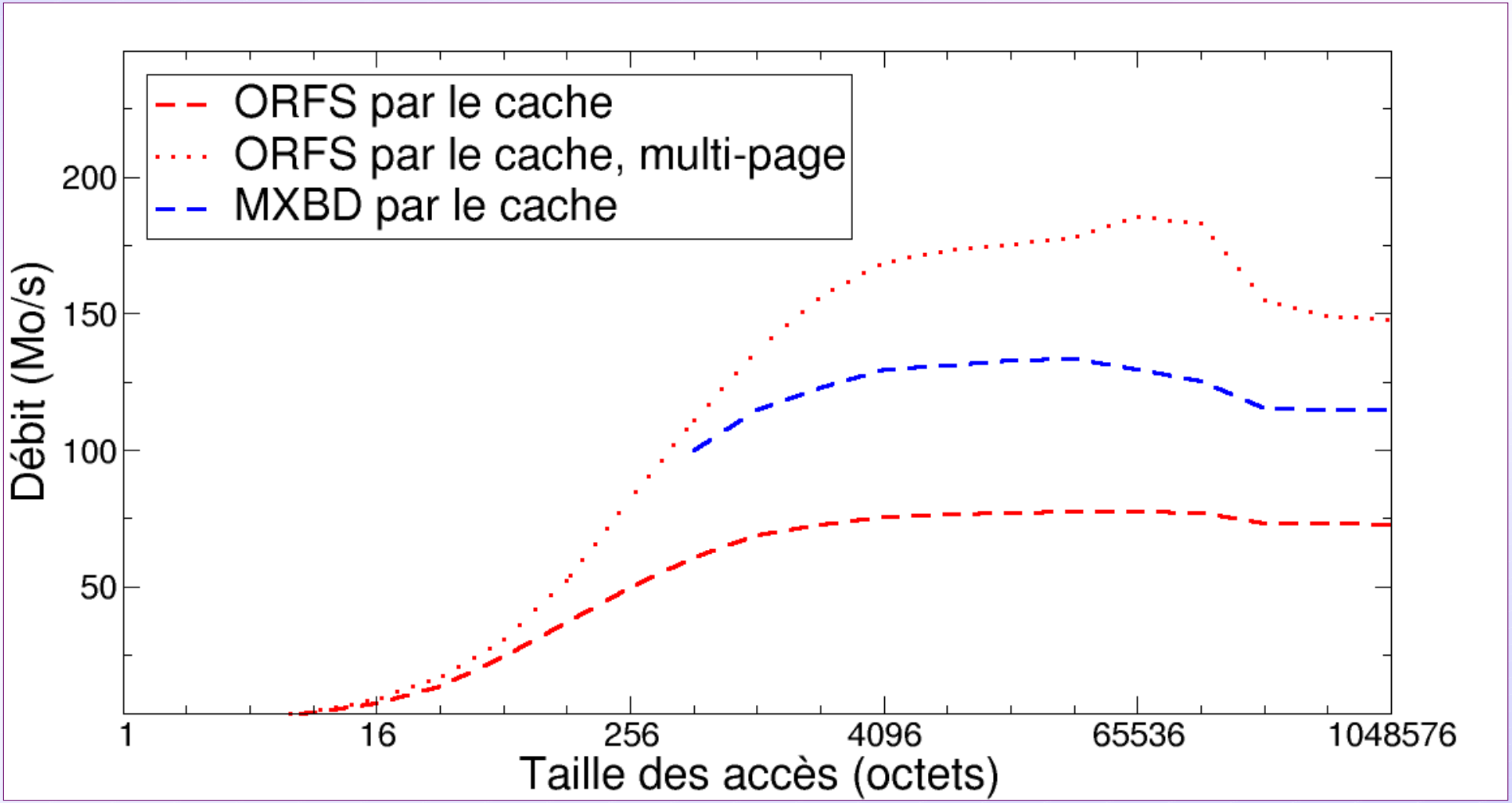
Performances des accès aux fichiers distants sur MX



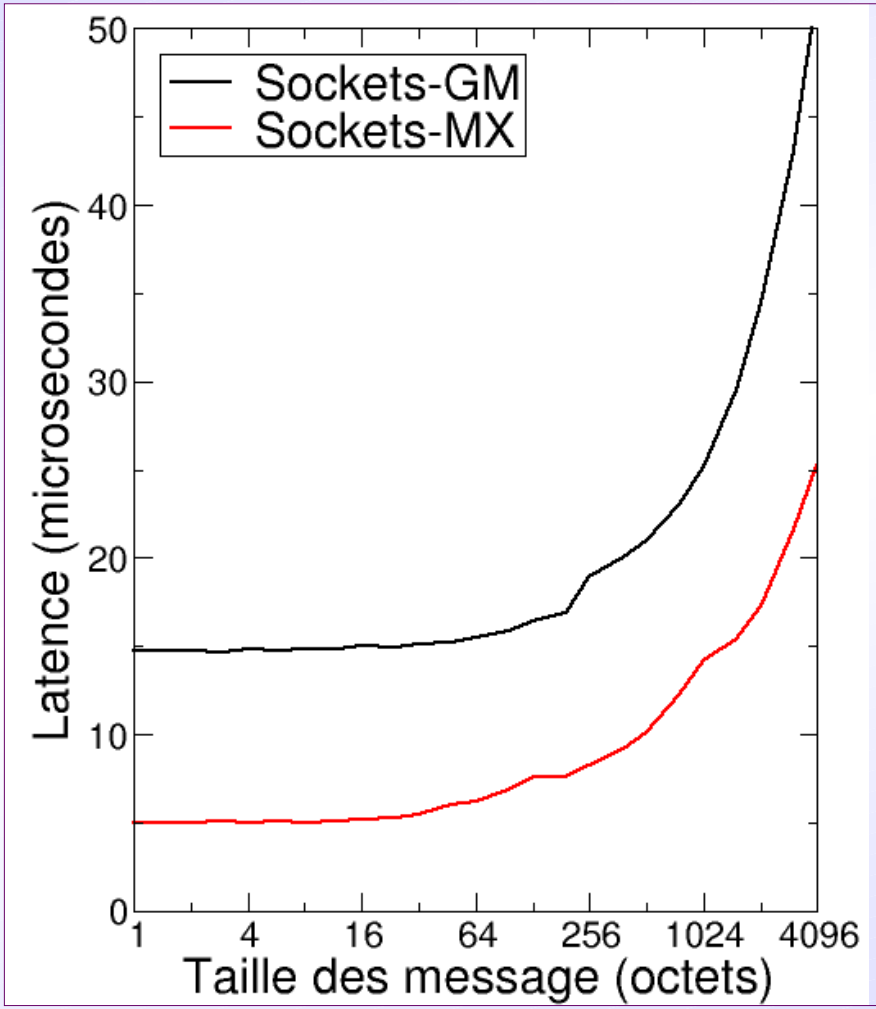
Impact des primitives vectorielles sur les accès par le cache

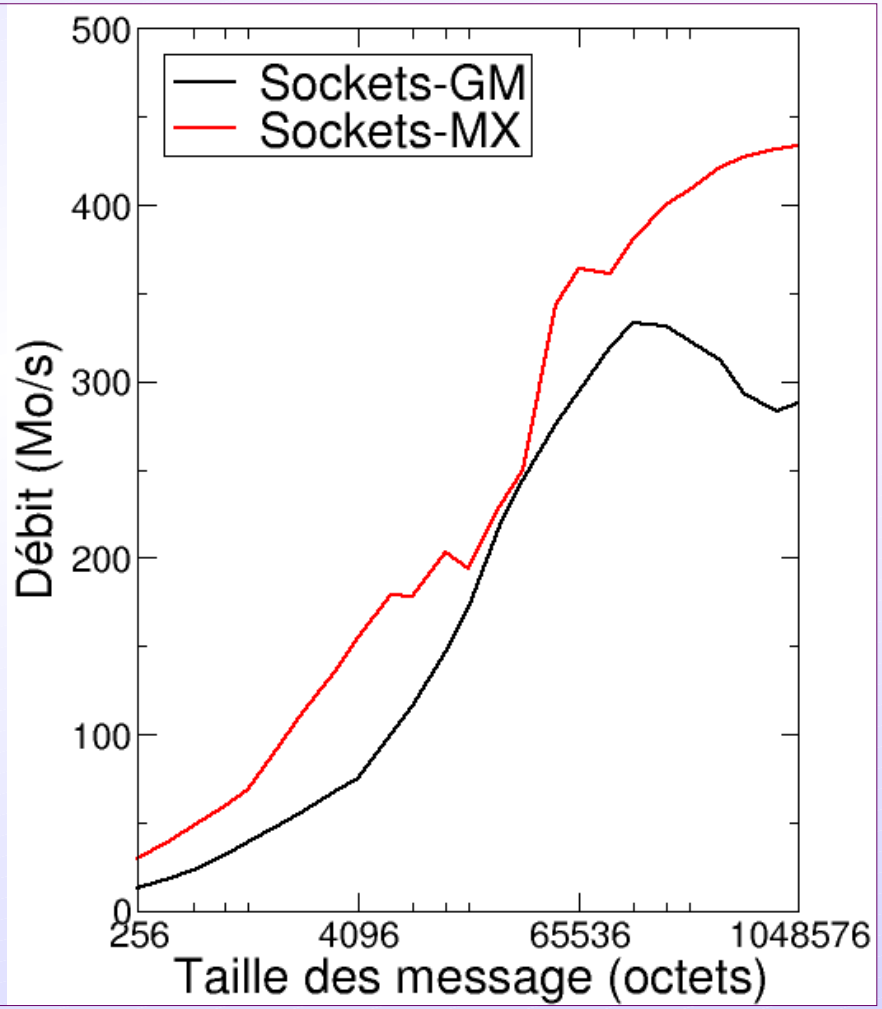


Impact des communications pipelinées sur les accès par le cache



Performances des Sockets zéro-copie sur MX





Plan de l'exposé

- Contexte
- Étude préliminaire
- Analyse des problèmes
- Propositions pour une interface noyau adaptée
- Conclusion et perspectives

Conclusion

- Gain important apporté par réseaux rapides
  - Jusqu'à 95% du débit réseau
- Besoins complexes en transfert de données
  - Très différent des communications en espace utilisateur
  - Nécessité d'adapter l'interface de programmation
    - et le système d'exploitation

Conclusion (2/2)

- Propositions pour une interface de programmation adaptée aux besoins en transfert de données
  - Intégrée à Myrinet MX
- Validation de l'utilisation de MX
  - Performances et facilité d'utilisation
    - Différentes méthodes de stockage distribué
    - Protocole Sockets zéro-copie

Publications

- Goglin, Prylli. *Performance Analysis of Remote File System Access on a High-Speed Local Network.* **CAC'04 Workshop, IEEE IPDPS Conference**. Santa Fe, Avril 2004.
- Goglin, Prylli. *Transparent Remote File Access Through a Shared Library Client.* **PDPTA**. Las Vegas, Juin 2004. Volume 3, pages 1131-1137.
- Goglin, Prylli, Glück. *Optimizations of Client's side communications in a Distributed File System within a Myrinet Cluster.* **HSLN Workshop, IEEE LCN Conference**. Tampa, Novembre 2004. Pages 726-733.
- Goglin, Glück, Primet, Mignot. *Accès optimisés aux fichiers distants dans les grappes disposant d'un réseau rapide.* **RENPAR'16**. Le Croisic, Avril 2005.
- Goglin, Glück, Primet. *An Efficient Network API for in-Kernel Applications in Clusters.* **IEEE Cluster 2005**. Boston, Septembre 2005.

Perspectives

- Mesure de l'impact combiné de nos travaux et des stratégies habituelles (parallélisation, ...)
  - Mise en œuvre de PVFS ou Lustre sur MX
- Poursuite des travaux avec Myricom
  - Primitives vectorielles de communication
    - Accès par pages ou blocs multiples
  - Notifications réseau par l'interface standard d'entrées-sorties
    - Gestion conjointe des événements disque et réseau

Perspectives (2/3)

- Support générique pour les réseaux rapides dans les systèmes d'exploitation
  - Travaux avec Quadrics, InfiniBand et les développeurs du noyau Linux
- Généralisation à Ethernet 10G
  - RDMA et TOE présentent des problèmes similaires
    - Support générique pour ces technologies et celles des réseaux rapides

Perspectives (3/3)

- Généralisation aux grilles de calcul
  - Protocole unique optimisé en courte et longue distance
    - Sockets zéro-copie sur les grappes
    - Transition transparente
  - Stockage dans les grilles
    - Bonne interaction entre réseau et stockage