

Soutenance de DEA

Documents Multimédia, Images et Systèmes d'Informations Communicants

Intitulé :

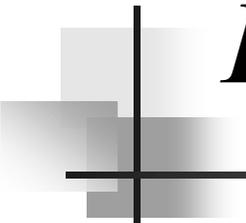
Intégration de la fonction de proxy cache aux routeurs actifs

Réalisé par : Guebli Sid Ali

*Encadré par : Jean Marc Pierson
Laurent Lefèvre*

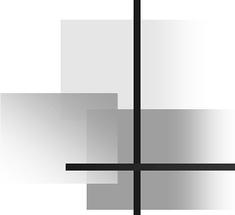


09 Juillet, 2003



Plan de la présentation

- *Introduction & Motivations*
- *Systemes de coopération*
- *Réseaux actifs*
- *Modèle proposé*
- *Implémentation*
- *Discussion*
- *Conclusion et perspectives*



Introduction

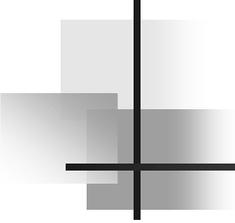
- Web est devenu l'application la plus utilisée sur Internet.
- Accroissement exponentiel du trafic www.
⇒ trafic redondant dans le réseau.

Solution :

Utiliser des systèmes de cache pour :

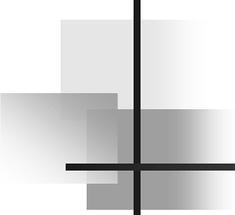
- Réduction du trafic réseau ;
- Réduire le chargement des serveurs occupés;
- Réduire le temps de latence des clients;

De meilleures performances en utilisant la coopération de caches (protocole de communication inter cache)



Introduction (2)

- Limite des réseau actuelle à déployer de nouveau protocole.
⇒ réseaux actifs (ensemble de nœuds actifs).
- L'idée est d'introduire une certaine intelligence dans le traitement des paquets traversant le réseau.
- Respecter les contraintes imposées par les noeuds actifs.
- Réduire le nombre de paquets qui circulent dans le réseau.

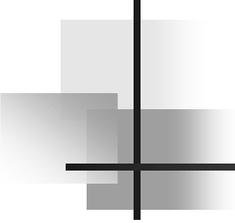


Systeme de cache cooperatifs

- ✓ Plusieurs cache distribués coopèrent pour partager les ressources entre eux.
- ✓ Un protocole de communication inter-cache.

- ✓ Grands systèmes de caches coopératifs :
 - Geographical Push :
 - Hierarchical : plusieurs niveaux.
 - Adaptive : groupes de caches forment des mailles (mesh).

Quel est le meilleur protocole de communication (ICP, cache digest, Summary cache, ...) ?



Les Réseaux actifs

- Domaine récent et en pleine expansion.
- Permet Augmenter la flexibilité et l'innovation dans les réseaux.
- Des mécanismes qui accélèrent le déploiement de nouveaux protocoles et services.
- Un nœud actif (routeur, proxy, passerelle, ..) :
 - rôle* : Effectuer les opérations de routage classique et des traitements sur les paquets actifs les traversant.
 - contraintes* : capacité de traitement et de stockage limitée.

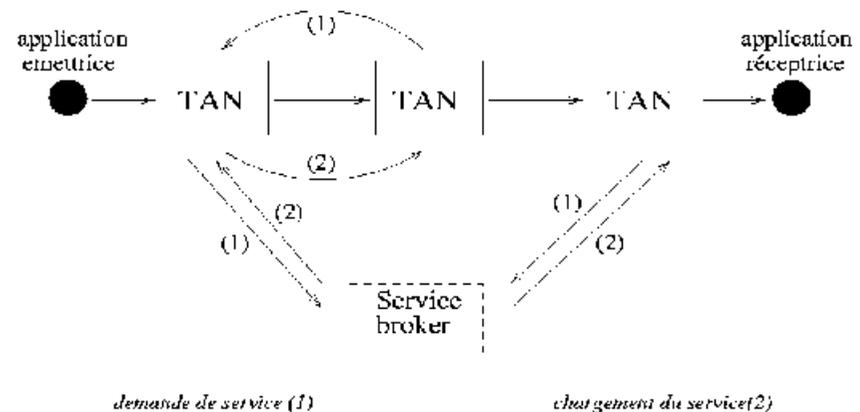
Les Réseaux actifs (2)

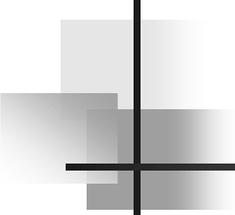
- Il existe deux approches :
 - L'approche *paquet actif* (intégrée);
 - L'approche *noeud actif* (discrète).

Déploiement de services : (approche discrète)

- ✓ Injection de nouveaux services dans le réseau actif.
- ✓ Deux mécanismes de déploiement :

- 1- Service Broker;
- 2- De voisin en voisin.





Modèle proposé

- ✓ Système de caches coopératifs sur des équipements actifs.
- ✓ Les fonctions essentielles du système de cache :
 - ***Découverte*** : comment localiser les objets cachés;
 - ***Délivrance*** : comment délivrer les objets aux caches voisins.
- ✓ Structure hiérarchique des caches (parent + fils).
- ✓ Données de coopération (tables miroirs).
- ✓ Système basé sur des services qui permettent la communication entre les caches.

Modèle proposé (2)

Choix de l'emplacement des caches :

- Caches fils : (prés des clients & disponibilité).
- Caches parents : regrouper une communauté de caches fils.

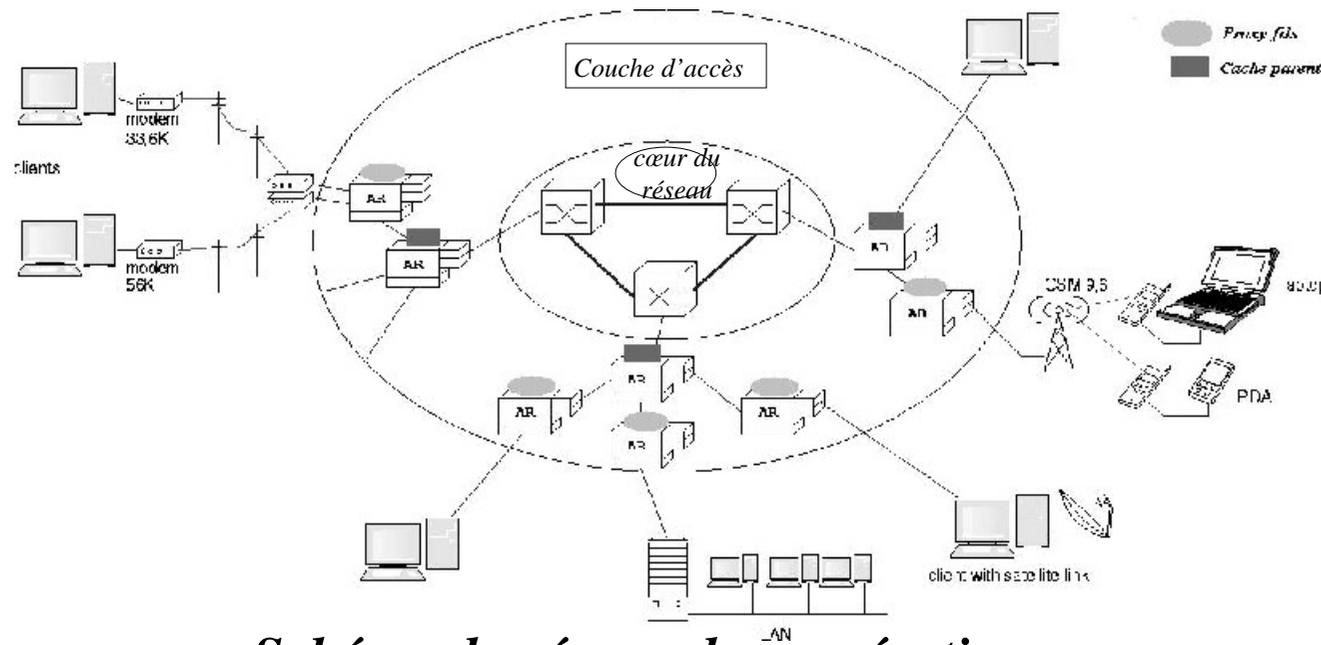
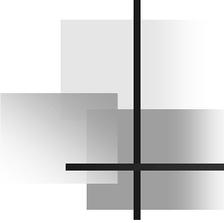


Schéma du réseau de coopération.



La table miroir

- ✓ Structure d'échange des informations de coopération.
- ✓ Utiliser pour la localisation des documents.
- ✓ Reflète le contenu de chaque cache.
- ✓ Caractéristique : codification très compacte.
utilisation de la technique de *bloom filter*.
- ✓ Des mécanismes pour l'envoi au parent.

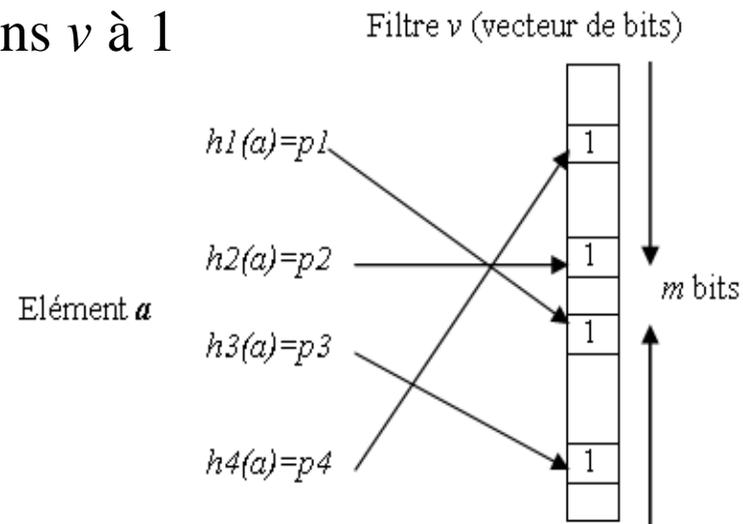
La table miroir (2)

Le bloom filter

- ✓ Représenter un ensemble de n éléments $E = \{a_1, a_2, \dots, a_n\}$ dans un tableau de bits v de taille m .
- ✓ k fonctions de hachage indépendantes $h_1, h_2, \dots, h_k \in [0..m]$
- ✓ Pour chaque élément a de E calculer $\{h_1(a), h_2(a), \dots, h_k(a)\}$
- ✓ Mettre les bits correspondants dans v à 1
- ✓ L'intérêt de cette codification.
- ✓ Problème de *faux Hit*.

$$p_{err} \approx \left(1 - e^{-\frac{kn}{m}}\right)^k$$

Guebli Sid Ali

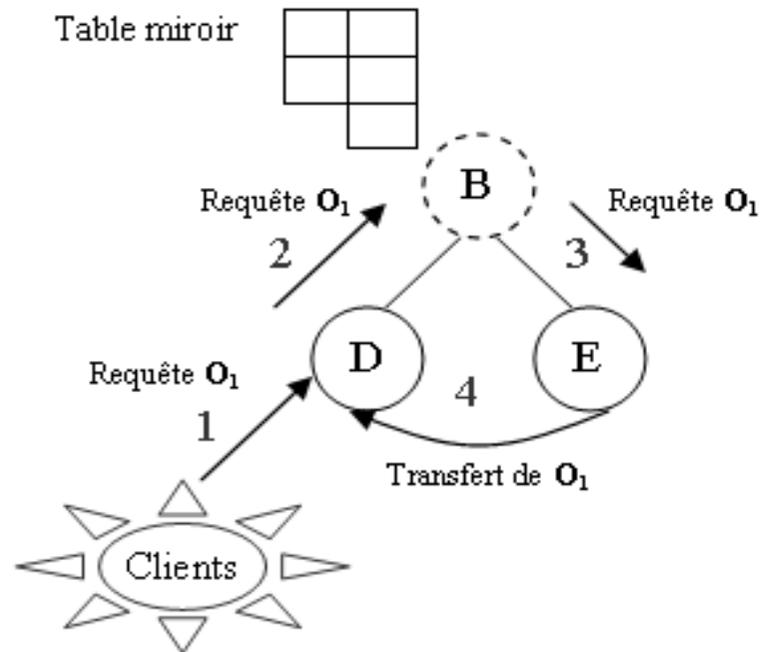


Mécanisme de communication inter-caches

- ✓ Sous l'administration d'un cache superviseur (parent).
- ✓ Basée sur les données la *table miroir*.
- ✓ La *découverte* :
 - Garder l'image de chaque cache fils chez la parent;
 - Consulter les tables miroirs pour rediriger les requêtes;
 - Equilibrage de charge des caches (stratégie LRU).
- ✓ La *délivrance* :
 - Transfert pair à pair;
 - Moins de charge sur les caches superviseurs.

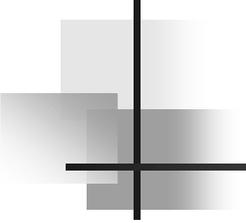
Mécanisme de communication inter-caches (2)

Exemple :



✓ Problème de consistance des tables miroirs.

⇒ Mécanismes de *mise à jour* de ces tables.



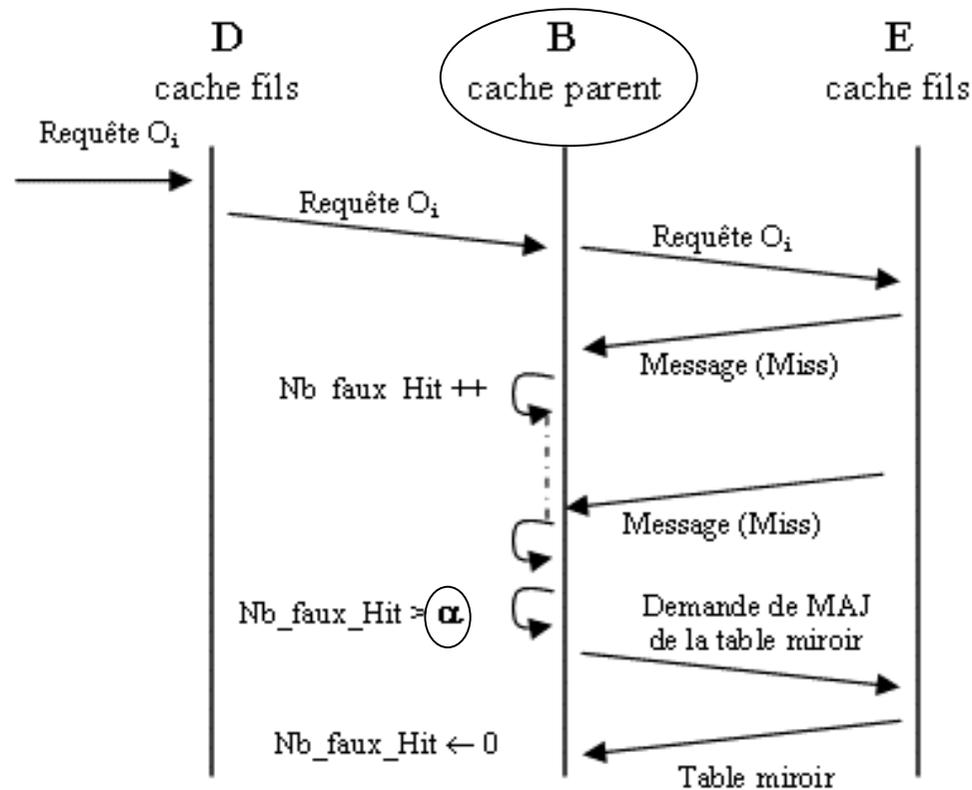
Mise à jour des tables miroirs

- ✓ Mécanisme pour garder la consistance des tables miroirs.
- ✓ Deux méthodes de mise à jour.

Paramètrée par des seuils de revalidation :

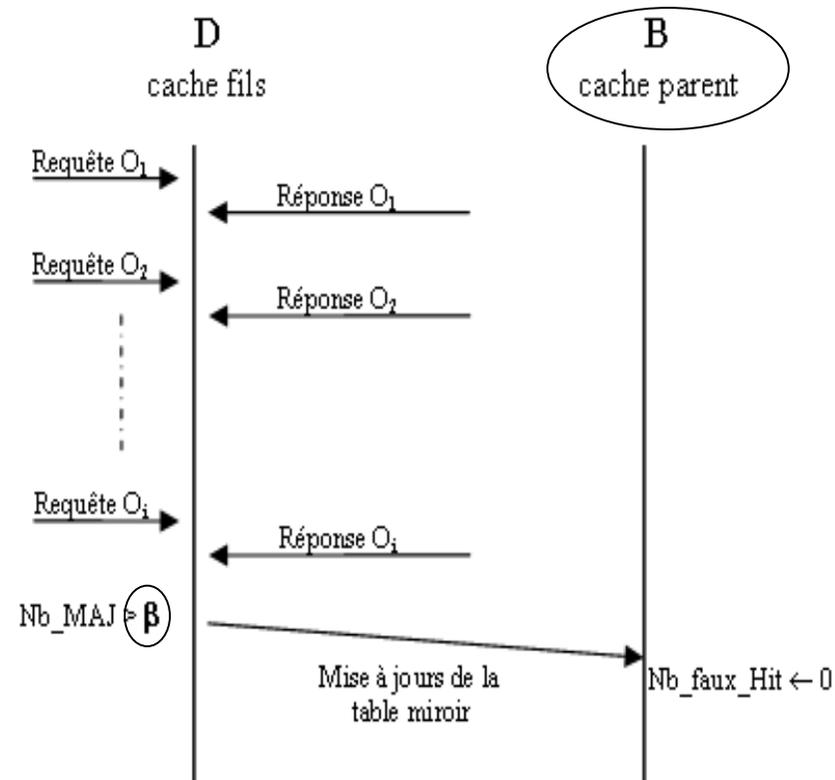
- α : côté du cache parent (nombre de Faux Hit);
- β : côté du cache fils (nombre d'opérations dans le cache).

Mise à jour des tables miroirs (2)



Du côté du parent

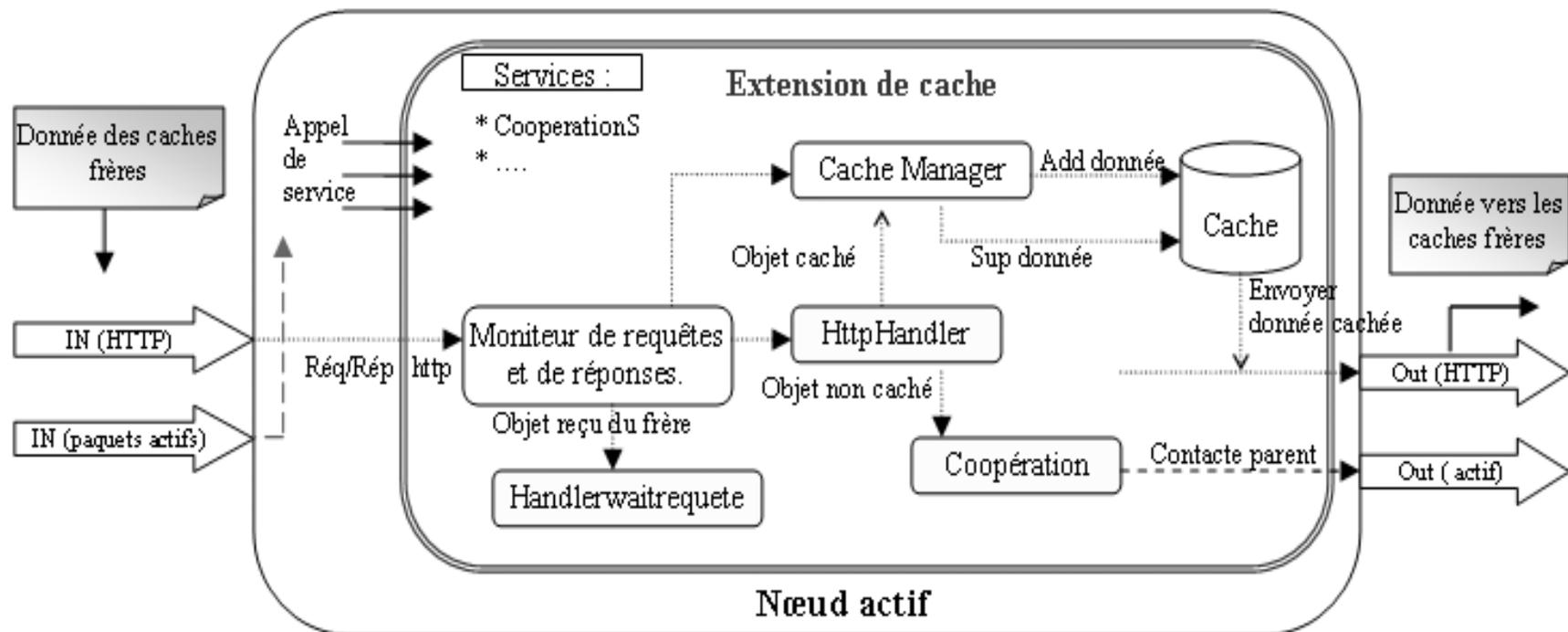
Mise à jour des tables miroirs (3)



Du côté du fils

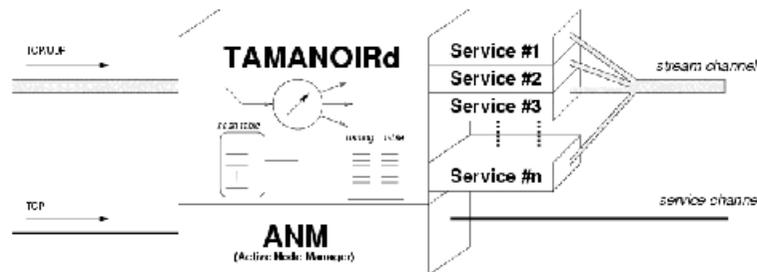
Architecture fonctionnelle d'un cache fils

- ✓ Basée sur des services légers (CoopérationS ,...).
- ✓ Schéma fonctionnel d'un nœuds fils:



Implémentation

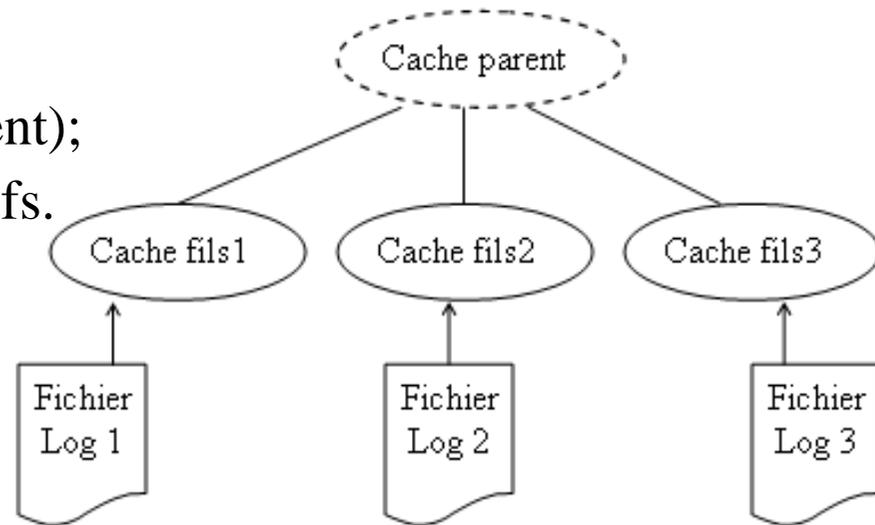
- Utilisation de la plate forme TAMANOIR (équipe RESO d'INRIA). Chaque nœud est composé de :
 - ✓ TAMANOIRd : démon qui tourne sur un nœud actif TAN (Tamanoir Actif Node).
 - ✓ ANM (Active Node Managment) : mettre à jour la table de routage locale, ainsi que télécharger de nouveaux services.



- ✓ Implémentations des différents services de *coopération*.

Expérimentations

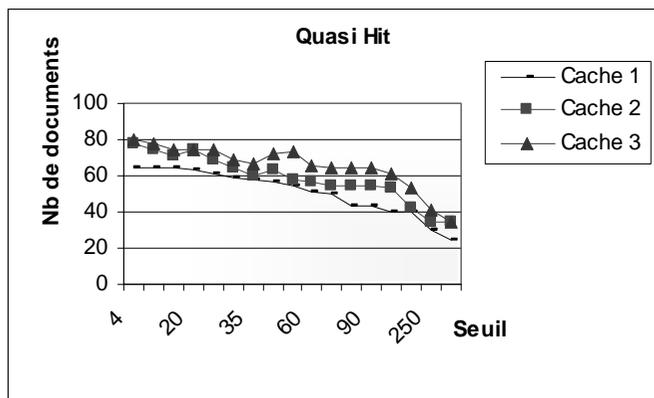
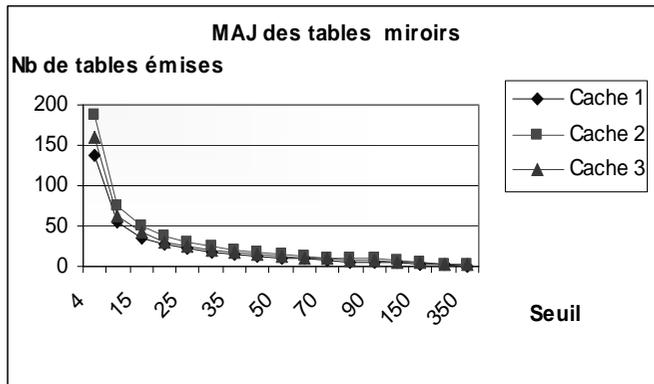
- ✓ Architecture de simulation:
 - 1 cache coordinateur (parent);
 - 3 proxies caches coopératifs.



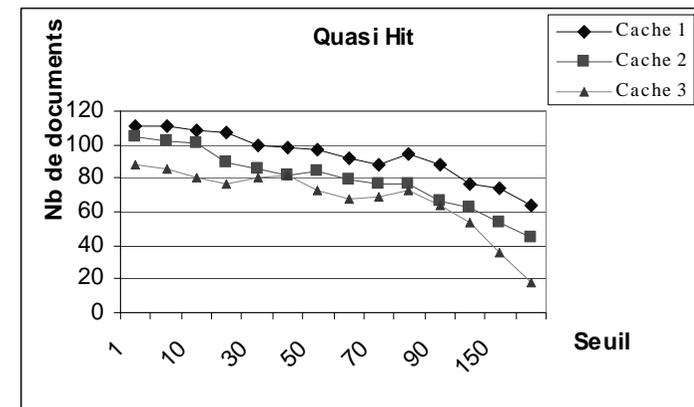
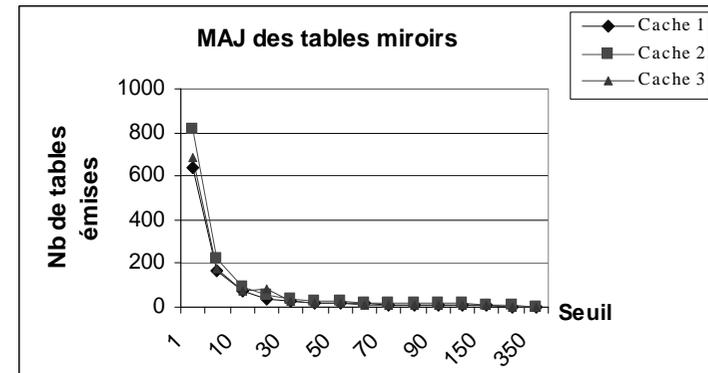
- ✓ Données de simulation
 - 4 fonctions de hachage;
 - taille du filtre $m=8000 \text{ bits} \approx 0,97 \text{ kilo octets}$;
 - Seuils de revalidation $\alpha = \beta$.
 - 1000 requêtes par proxy cache;
 - 1766 documents différents.

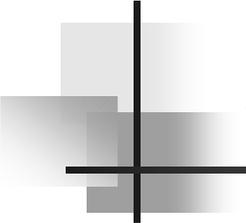
Expérimentations (2)

Caches infinis



Caches finis



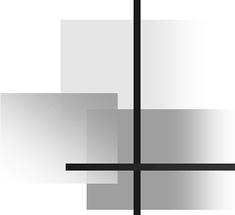


Discussion

- ✓ Permettre une localisation rapide d'un document dans une communauté de caches coopératifs.
- ✓ Limite le nombre de messages échangés entre les caches coopératifs pour récupérer un document.
- ✓ L'échange des tables miroirs ne se fait qu'à des moments opportuns.
- ✓ Possibilité de surcharger les proxies caches fils par les demandes des autres caches voisins.

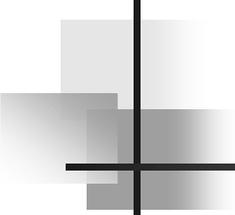
Solutions

- ☑ Ajouter du cache au niveau des caches coordinateurs;
- ☑ Caches fils travaillant dans un contexte « Best effort ».



Conclusion

- ✓ Extension de la capacité d'un nœud actif (intelligence de traitements des requêtes).
- ✓ Proposition d'un modèle de système de cache coopératifs, en respectant les contraintes des nœuds actifs.
- ✓ Organisation hiérarchique du système + définition d'un protocole de communication inter caches.
- ✓ Caractéristiques : *Localisation, Délivrance*.
- ✓ Implémentation sous la plate forme TAMANOIR.



Perspectives

- ✓ Une étude expérimentale plus poussée pour affiner les paramètres du système et leurs interdépendances.
- ✓ S'attaquer au problème des pages dynamiques, connu sous le nom de « cache actif ».
- ✓ Rendre le déploiement des caches dynamique sur les nœuds actifs.



Questions ?

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.