UNIVERSITÉ
CAEN
NORMANDIE

MM
*22

La gestion du cache

Anne Garnavault

Didier Cadiou

Jean-Marie Dumont

MOODLEMOOT CAEN*2022

UNIVERSITÉ CAEN NORMANDIE

Sommaire

- Le contexte
- Quelques éléments de définition
- Le cache quoi ca sert?
- Améliorer la gestion du cache

MOODLEMOOT CAEN*2022

UNIVERSITÉ CAEN NORMANDIE

CONTEXTE

- **AVANT MARS 2019**
- 2 serveurs en Load Balancing
- un serveur memcached sur chacun
- une gestion du cache mal maitrisée mais pas trop de soucis de performances

- **APRES**
- un usage des plateformes multiplié par 10
- 7 serveurs en //
- pas de modification des définitions de cache
- PLANTAGES REGULIERS POUR LES UTILISATEURS et manque de fiabilités des données (achèvement)

definition cache

- **Cache : données stockée temporairement afin d'accélérer l'application et d'améliorer les performances**
- Lors d'une requête http (client) , si les données en cache existent, elle sont renvoyées en priorité (données déja calculées)
- durée de validité en fonction des besoins (stockage temporaire)
- doit être le plus rapide possible pour améliorer encore les performances
- nécessité de purger le cache en cas de mise à jour importante

Vocabulaire - 1

- **Définition de cache** désigne une famille d'éléments à mettre en cache (par exemple les données d'appartenance d'utilisateurs à des groupes, ou les achèvements d'activités...). Elles peuvent être de 3 types :
  - **Application** : pour les données non spécifique à l'utilisateur
  - **Session** : pour les données de l'utilisateur, le temps de la session (temps pendant lequel l'utilisateur est connecté à Moodle) c'est-à-dire les données qui vont se retrouver au moment de l'exécution dans la variable PHP : **$_SESSION['SESSION']→cachestore_session**
  - **Requête** : pour les données de l'utilisateur, le temps de chacune des requêtes http exécutée pendant la session (ne pas confondre avec les requêtes SQL).

Vocabulaire - 3

- **Entrepôts de cache** c'est un type dispositif qui permet de stocker des données de cache (cache fichier, serveur Memcached, Redis...) - Idéalement stocké localement pour éviter les temps d'accès liés au réseau (pas toujours faisable) .

- **Instances d'entrepôt** il est possible d'avoir plusieurs entrepôts du même type (par exemple plusieurs caches fichiers avec différentes localisations). Chacun s'appelle une *instance*. Chaque instance se distingue par une configuration propre.

MOODLEMOOT CAEN*2022 | UNIVERSITÉ CAEN NORMANDIE

Vocabulaire - 3

- **MUC** (Moodle Universal Cache) est le dispositif Moodle qui permet un paramétrage fin de la configuration du cache. On y trouve notamment le choix des entrepôts de cache utilisés pour différents types de contenu (appelé “définition de cache”). Tous les éléments mis en cache par Moodle ne sont pas configurable dans le MUC, mais il y a quand même une soixantaine de définitions concernant des plugins ou des éléments du coeur que l'on peut configurer.

- **Mapping** ou **correspondance** : désigne le fait de définir l'instance d'entrepôt qui sera utilisé par une définition donnée.

Voir https://docs.moodle.org/dev/The_Moodle_Universal_Cache_(MUC)#Cache_types

serveur de cache

- **Un serveur de cache** (REDIS, MEMCACHED) : systeme de cache hébergeant un base de données (non relationnelle / paires de clé+valeurs) et qui fonctionne en mémoire vive (pour la performance)

RamDisk (tmpfs)

- montage d'un espace en mémoire vive (et non sur un disque)
- le serveur le considère comme un stockage disque et l'utilise comme tel (gestion dossiers/ fichiers)
- à l'arrêt du serveur, tout le contenu en RAMdisk disparait (mémoire vive)
- accès hyper rapide

Les réglages par défaut

config.php

$CFG->cachedir ($CFG->dataroot/cache (moodledata))

CACHE D'APPLICATION

$CFG->localcachedir ($CFG->dataroot/cache) tous les éléments qui peuvent être mis sans risque en local sur serveur et ne font pas l'objet d'un définition MUC

Dans les 2 cas, peu performant (surtout si montage NFS) mais intégrité des données respectées

MUC

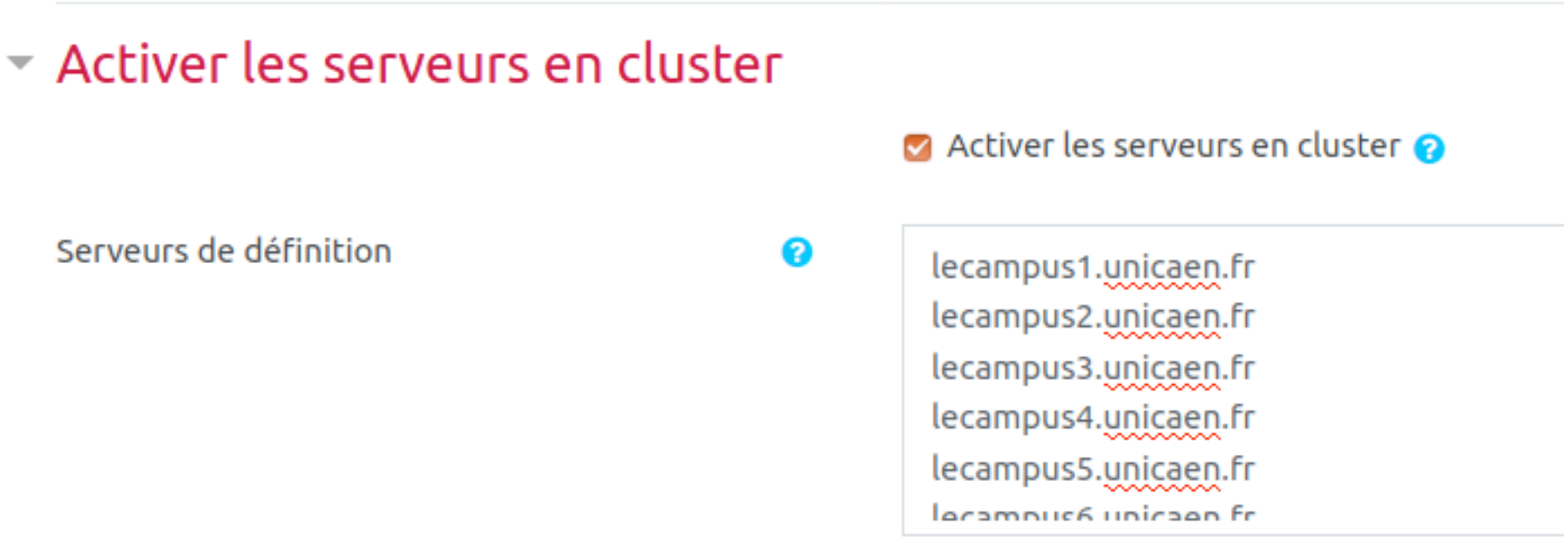
Paramétrable via **Administration du Site > Plugins > Cache > Configuration**
Ces réglages sont écrit par Moodle dans le fichier : *moodledata/muc/config.php*

La config du MUC n'est pas stockée en base de données.

(à noter qu'il peut être très pratique de supprimer une partie dece fichier pour réinitialiser les définitions)

les serveurs en clusters et le cache : topologie 1

- TOPOLOGIE 1 : un serveur memcached sur chacun des serveurs



MOODLEMOOT CAEN*2022

UNIVERSITÉ CAEN NORMANDIE

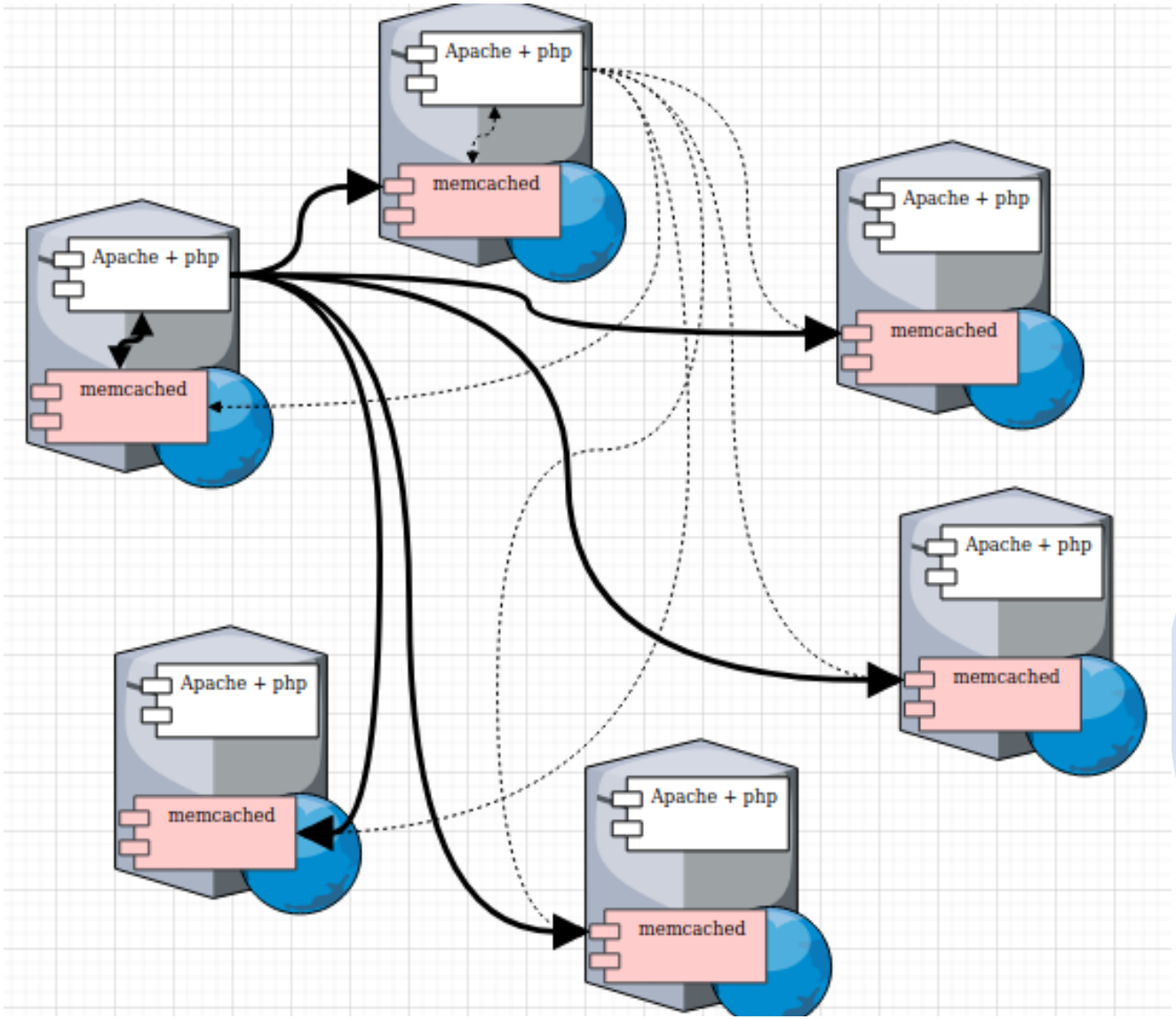
TOPOLOGIE 1

Lecture: chaque worker va lire les données en cache sur son instance memecached

Ecriture : sur toutes les instances (y compris la sienne)

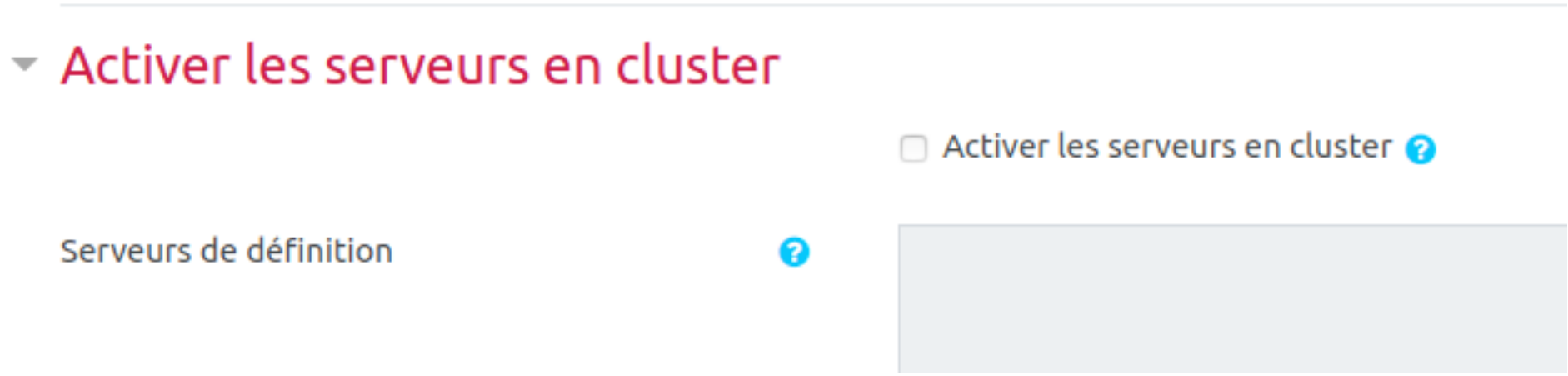
Ecriture moins fréquentes que la lecture = cache plutôt local

SI UN WORKER « TOMBE » et qu'une écriture est en cours => timeout de 20S => BLOCAGE



les serveurs en clusters et le cache : topologie 2

- TOPOLOGIE 2 : un serveur REDIS ou memcached externalisé



-

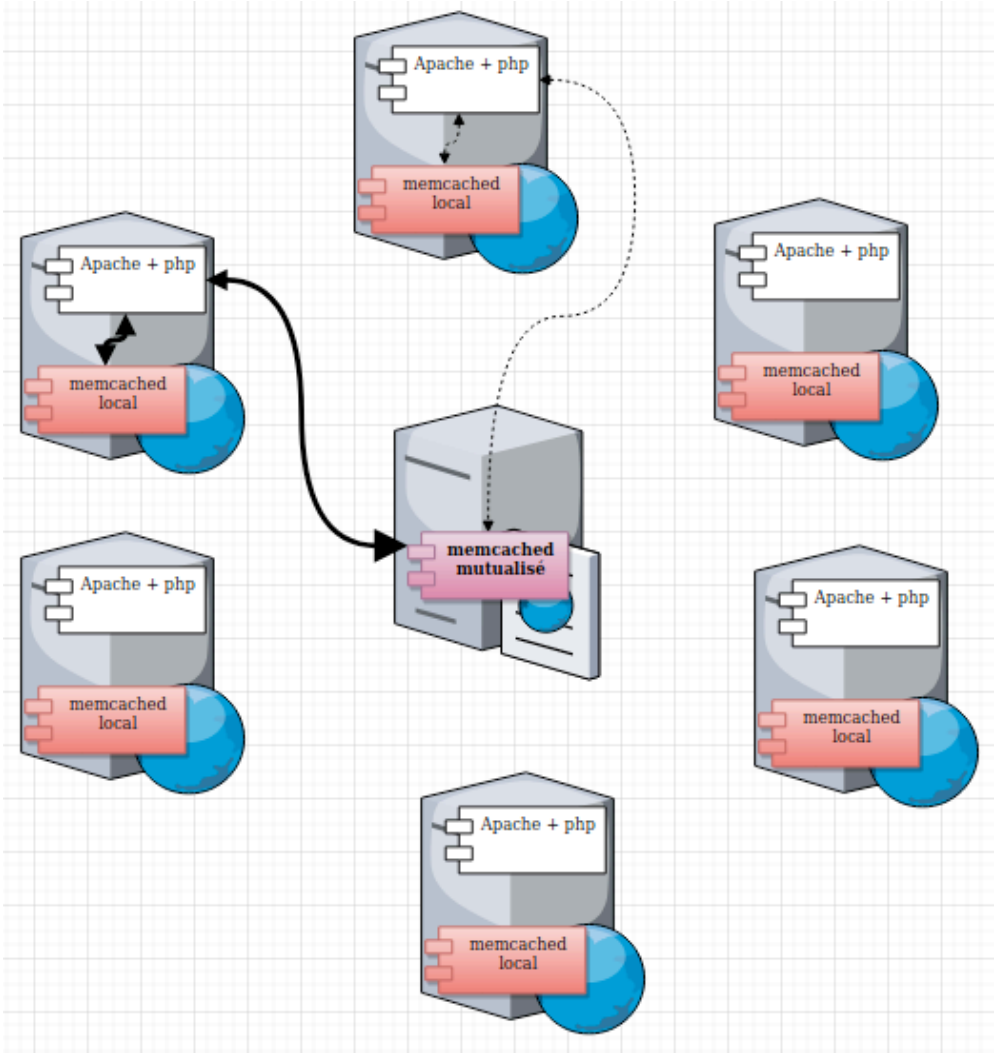
MOODLEMOOT CAEN*2022

UNIVERSITÉ CAEN NORMANDIE

TOPOLOGIE 2

Gérer différemment ce qui peut être mutualisé ou pas (dans la configuration - voir diapo suivante)

Vigilance sur le dimensionnement de l'instance mutualisée qui va concentrer le trafic de tous les worker



partage ou non?

- Distinguer ce qui partageable ou non (local) dans la configuration
- Accélérer le traitement du cache local via un montage tmpfs (traitement rapide en RAM)
    - à gérer dans le config.php apres création des montages à la racine de chaque worker ($CFG->localcachedir et $CFG->cachedir)
- Tout ce qui doit être partagé entre tous les workers est mis en cache sur un serveur MEMCACHED ou REDIS séparé , en général caches d'application (**=> *peut utiliser un entrepôt local = non***)
- Tout ce qui peut être partagé est traité via tmpfs (**=> *peut utiliser un entrepôt local = oui)***

MOODLEMOOT CAEN*2022

UNIVERSITÉ CAEN NORMANDIE

les caches de sessions

- **Distinguer « session » de l'utilisateur » et « cache de session »**
- **session utilisateur** -> point de configuration, ***dbsessions*** (dans *Administration* > *Serveur* > *Gestion des sessions*) indique si les sessions sont stockées en base de données ou non. Dans les versions récentes de Moodle, *dbsessions* est désactivé par défaut, dans ce cas elles sont stockées dans un entrepôt fichier (*$CFG→dataroot*/sessions).
- **cache de session** -> regroupe toutes les données qui accompagnent les sessions et qui doivent/peuvent être mises en cache . Exemples : catégories de calendrier qu'un utilisateur peut consulter, information d'abonnement app mobile, données avant inscription, requêtes sur catégories ce notes, résultats de recherches d'éléments taggués ou sélections d'utilisateurs, etc.
- Aucune n'est locale. Elles doivent passer par le serveur REDIS/ MEMCACHED ou configuration par défaut (moodledata)
- Le fait d'utiliser une répartition de charge sur HAPROXY en mode "sticky session" N'EST PAS DU TOUT SUFFISANT pour garantir la bonne cohérence des données en cache et ainsi prévenir tout dysfonctionnement temporaire. (malgré ce qui dit dans la documentation https://docs.moodle.org/38/en/Caching#Cache_types_and_multiple-server_systems)

les caches de requêtes

- Requêtes http le temps d'une session utilisateur
- Par nature, cache LOCAL - aucun besoin de le mutualiser
-

à retenir

Principes généraux:

- cache en mémoire : toujours (tmpfs ou memcahed/redis) = améliore les performances (proscrire cache sur disque)
- cache local quand c'est possible
- cache partagé = la fiabilité des données si serveur en cluster
- importance de la configuration et lecture des définitions
- ne pas mettre ses oeufs dans le même panier (= solutions de secours temporaires)

Sessions : pas de stockage en BD

ELEMENT CLÉ DE LA PERFORMANCE DE LA PLATEFORME