
numad

Service en userspace qui fournis des conseils de placement et de gestion de processus pour utiliser efficacement les CPU et la mémoire dans le système avec une topologie NUMA

numad est un service qui monitor la topologie NUMA et l'utilisation des ressources. Il tente de localiser les processus pour la localité NUMA, l'affinité, en ajustant dynamiquement pour changer les conditions système. numad fournis également un guide pour assister la gestion des applications avec une liaison manelle des ressources CPU et mémoire.

OPTIONS

- C 0|1** Cette option contrôle si numad traite le cache de fichier inactif comme mémoire disponible. Par défaut, il peut compter le cache de fichier inactif comme mémoire libre.
- d** mode debug
- H <THP_scan_sleep_ms>** Définis l'intervalle de scan transparent hugepage en milliseconde.
/sys/kernel/mm/transparent_hugepage/khugepaged/scan_sleep_millisecs est généralement de 10000ms par l'os. Le défaut est changé par numad à 1000ms vu que ça aide le service hugepage à être plus agressif quand il déplace la mémoire entre les nœuds. à 0, numad laisse la valeur du système. 100ms peut améliorer les performances des processus à forte charge qui utilisent de nombreuses transparent_hugepages.
- i <[min_interval :] max_interval>** Interval de temps d'attente entre les scans système. Défaut : 5 (min) et 15 (max) secondes.
max_interval à 0 cause le service à quitter. Une valeur plus grande diminue la charge de numad, mais diminue la réponse les charges changeantes.
- K 0|1** Cette options contrôle si numad conserve entrelacement de la propagation de mémoire au travers des nœuds NUMA, ou tente de fusionner la mémoire entrelacée dans les nœuds NUMA locaux. Le mode par défaut est de fusionner la mémoire entrelacée C'est approprié pour localiser les processus dans un sous-jeu des nœuds NUMA du système. Si vous lancez une grosse application simple instance que alloue de la mémoire entrelacée parce que la charge aura des motifs d'accès mémoire imprédictibles, les résultats seront meilleur avec -K 1 pour instruire numad de conserver la mémoire entrelacée distribuée.
- l <log_level>** Définis le niveau de log. défaut : 5
- m <target_memory_locality>** Définis le seuil de localité mémoire désiré pour stopper le déplacement de mémoire. numad peut arrêter de traiter la mémoire quand plus de ce pourcentage de mémoire du processus est déjà localisé dans le ou les nœuds cible. le défaut est 90%. numad localise fréquemment ce seuil mais ne le fait pas nécessairement. à 100%, numad tente en permanence de déplacer la mémoire que le kernel n'arrivera jamais à déplacer.
- p <PID>** ajoute le pid à la liste d'inclusion des processus à considérer pour la gestion, si le processus utilise déjà des ressources significative. Peut être spécifié plusieurs fois au démarrage, mais une fois lancé, un seul pid peut être ajouté à la liste d'inclusion par invocation de numad. Utilisé avec -S pour contrôler précisément le périmètre des processus que numad peut gérer. Noter que ce processus spécifié ne sera pas nécessairement géré activement sauf s'il rencontre le seuil de numad, qui est actuellement de 300Mo et 1/2 CPU.
- r <PID>** Supprime le pid des liste d'inclusion et d'exclusion des processus. une fois le service démarré, seul un PID peut être supprimé par invocation de numad. Avec -S, -p et -x pour contrôler précisément le périmètre des processus que numad peut gérer.
- R <CPU_LIST>** Spécifie une liste de cpu que numad devrait considérer comme réservé pour une utilisation non numad. Aucun processus ne sera lié aux CPU spécifiés par numad. Cette option est seulement effective quand numad démarre.
- S 0|1** Cette option contrôle si numad scanne tous les processus système ou seulement les processus dans la liste d'inclusion PID.
Scan tous les processus par défaut. -S 0 scanne seulement la liste d'inclusion PID . numad -S 0 -p <PID-1> -p <PID-2> -p <PID-3> limit le scan, et donc la gestion automatique NUMA, à seulement 3 processus.
- t <logical_CPU_percent>** Spécifie la valeur de ressource des CPU logique. Les threads hardware partagent généralement pluse de ressources, et ces CPU logiques ajoutent seulement une fraction de puissance CPU. Par défaut numad considère les CPU logiques à seulement 20% d'un cœur hardware dédié.

-
- u <target_utilization> Définis le pourcentage de consommation d'un nœud. Défaut : 85%. Diminuer cette valeur pour maintenir plus de ressource disponible dans chaque nœud. à 100, les performances systèmes sont améliorée. Il est possible d'aller jusqu'à 130 pour surcharger les CPU dans les nœuds, mais l'utilisation mémoire est toujours plafonnée à 100%.
 - v mode verbeux
 - w <NCPUS [:MB] > demande à numad les meilleurs nœuds NUMA pour lier une entité qui nécessite <NCPUS>. La quantité de mémoire est optionnelle, mais devrait normalement être spécifiée pour que numad recommande les nœuds NUMA avec la capacité CPU disponible et la mémoire disponible. La sortie de cette option est une chaîne qui contient une liste de nœud NUMA. Cette liste devrait être placé dans une variable (ex :NODES), et utilisé comme paramètre dans numactl -m \$NODES -N \$MODES ...
 - x <PID> Ajoute le PID à la liste d'exclusion de processus à blacklister de la gestion. Peut être spécifié plusieurs fois au démarrage, mais une seul fois ensuite par invocation de numad. avec -S permet de contrôle précisément le périmètre des processus que numad peut gérer.

Exemples

numad peut être lancé comme service système :

/usr/bin/numad

les autres invocation de numad peuvent être utilisé pour changer dynamiquement les options à chaud :

/usr/bin/numad -i0