

Enezis



**Résolution des problèmes de
temps de réponse**



Pourquoi Enezis ?

Enezis dispose d'un capital de 15 ans d'expérience dans le domaine des performances systèmes Unix (HP-UX, AIX et Solaris) et Windows, bases de données Oracle et MsSqlServer, baies de stockage HP et EMC. Auteurs de dizaines d'interventions sur ces sujets, le consulting d'Enezis travaille par exemple pour Hewlett Packard pour délivrer les formations consacrées aux performances et aux internes systèmes de HP-UX. Un livre sur les performances intégrées Unix/ Oracle/ baies de stockage est en préparation.

Résolution des problèmes de performances: essayer, essayer et essayer encore...

La méthode la plus employée dans la résolution des problèmes de performances consiste après une analyse souvent sommaire à « essayer » des changements de paramètres « pour voir » : augmentation des zones de mémoires allouées, changement de configurations système et stockage, etc. Cette méthode présente deux inconvénients majeurs : elle est tout d'abord généralement appliquée plus ou moins au hasard en faisant varier quelques paramètres connus, qui représentent une fraction des dizaines de variables configurables (bases de données, système d'exploitation, réseau, baies de stockage, couches logicielles applicatives), sans même évoquer les combinaisons et les interdépendances. Elle présente par ailleurs le défaut majeur de s'adresser à des réglages techniques (taux de réussite dans les différents caches par exemple) qui n'ont généralement qu'un rapport lointain, voire aucune relation, avec les temps de réponse des utilisateurs.

Certains « tuning tips » et des cours ou livres anciens consacrés aux performances versent d'ailleurs dans ce travers : tel « tuning guide » de bases de données donnera par exemple comme objectif d'obtenir un taux de réussite supérieur à 99% des ordres SQL dans le cache, tandis qu'un livre consacré aux performances Unix précise qu'il faut avoir un taux de réussite supérieur à 90% dans le cache de données des systèmes de fichier (« buffer cache »). On oublie dans le premier cas que nombre d'applications n'émettent jamais deux fois le même ordre SQL et bénéficient de très bons temps de réponse. Quand au second cas, comment un tel résultat peut il être possible même avec 32Go de RAM si les tailles des index parcourus se chiffrent en centaines de Go ? Pire encore, le changement de matériel (le fameux « upgrade ») est fréquemment utilisé afin de ne pas faire preuve d'inactivité. Combien parmi ces « upgrades » se sont-ils soldés par des résultats négligeables ?

Pour être efficace, l'approche doit être structurée

Les méthodes modernes d'analyses de performances sont 1) structurées et 2) approchent le problème des temps de réponse en ventilant les temps de réponse sur différents postes (CPU et « attentes »). La démarche consiste avant tout à identifier le goulot d'étranglement primaire, et éventuellement le goulot secondaire. Certaines interfaces de mesures (par exemple l'OWI – Oracle Wait Interface -), ou encore Glance UX pour HP-UX permettent de ventiler et de hiérarchiser les raisons d'attente (i.e. de non consommation de CPU). Cette interface de mesure n'existe en revanche pas par exemple sur Windows dans le performance monitor : il faut alors inférer les raisons d'attente en se servant de 1) règles simples issues de la théorie des files d'attente et 2) d'indicateurs particuliers comme par exemple dans Ms SqlServer.

L'approche intégrée

Il est illusoire de vouloir aborder les problèmes de performances en se cantonnant à l'étude de l'un des secteurs technique : stockage pour les E/S, bases de données, système, etc. L'ingénieur système peut voir une forte consommation de CPU sur un système et s'en inquiéter, là ou le DBA se félicitera car il a réussi à supprimer les autres postes sur lesquels les requêtes SQL passent du temps. Qui a raison ? Aucun a priori tant que les deux questions suivantes ne sont pas résolues : 1) les temps de réponses sont-ils bons ? Ils peuvent l'être si par exemple le temps « elapsed » (montre en main) est



égal au temps CPU 2) la consommation de CPU met elle en jeu l'évolutivité du serveur ? En d'autres termes, le système doit il recevoir des pics de charge supplémentaires ?

Cet exemple simple fait apparaître le hiatus qu'on trouve dans la plupart des sociétés : chaque domaine technique a ses propres objectifs, ses propres indicateurs qui ne correspondent que de façon éloignée aux problèmes posés, à savoir l'amélioration des temps de réponse.

Connaître

Mal connues, les performances sont souvent entourées d'un halo de mystère. La réalité est toute autre : la plupart des informations sont assez facilement disponibles à travers des livres, des cours ou des notes données par les constructeurs et les éditeurs. La difficulté est d'agréger ces informations hétérogènes dans un ensemble cohérent qui permette de comprendre les mécanismes en jeux.