

Relations entre SolidWorks, la carte mère, la mémoire de l'ordinateur et la carte graphique.

Ce petit document sans prétention a pour but de comprendre à minima le fonctionnement de l'ordinateur lorsque l'on utilise SolidWorks 2010 et/ou 2011.

Cela devrait permettre de cadrer un peu les usages et, éventuellement, aider dans le choix d'une configuration matérielle.

Du point de vue technique il faut déterminer comment les éléments suivants s'articulent.

- Le traitement parallèle ou non et dans quels cas (multithreading)
- La quantité de mémoire nécessaire
- Le fonctionnement de la CPU
- Les processeurs multi-cœurs
- SolidWorks
- Le fonctionnement de la GPU (Graphic Process Unit).
- Windows 7

Du point de vue pratique nous verrons aussi les différents types d'affichage des plus simples au plus sophistiqués.

Précautions prises avant le lancement du test

Tous les autres programmes de la station sont fermés (skype, firefox, etc...) auparavant pour ne pas perturber le test. Au lancement de SW, les modules Photoview et Simulation sont désactivés, puisqu'ils peuvent être chargés ou non au lancement.

Toutes les mesures sont réalisées sur deux ordinateurs différents (voir en annexe : les configurations utilisées pour le mini-test).

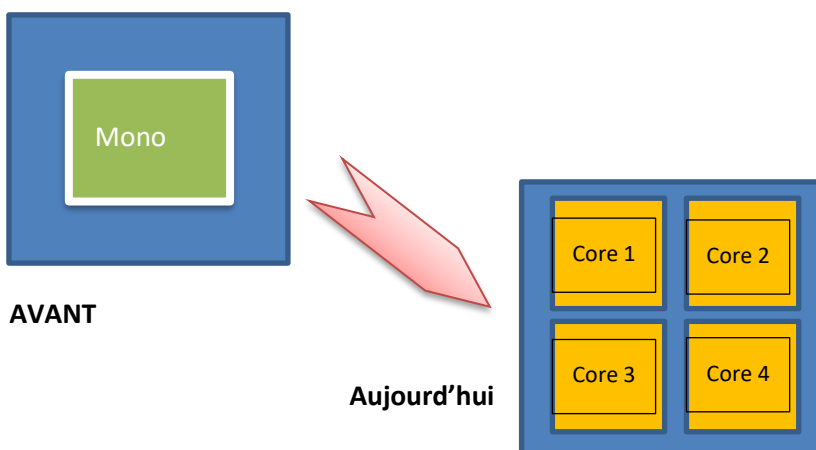
Les processeurs multi-cœurs

Autrefois il y avait un seul processeur. La miniaturisation aidant, les constructeurs ont mis sur un seul chips, deux, quatre processeurs ou plus qui seront appelés cœurs.

Comme dans les faits, il y a un seul processeur et pour éviter toute polysémie ils les ont appelés processeurs multi-cœur (dual core, quad core dans le jargon).

Ces cœurs peuvent travailler totalement indépendamment les uns des autres. En clair ils peuvent exécuter des programmes séparés, ou un seul programme mais en l'exécutant par morceaux indépendants. Seuls les multi-cœurs permettent de faire du traitement parallèle qu'il ne faut pas confondre avec l'OS multitâches.

Le chef d'orchestre de tout cet ensemble est bien sur l'OS (Windows) qui lui est multitâches.



Le traitement parallèle ? (multithreading)

Aujourd'hui beaucoup d'ordinateurs sont dits « multi-cœurs » de 2 à 6 cœurs ou plus. Mais il ne faut pas confondre les threads et les cores.

Avant que vous ne tombiez à la renverse tout raide, il faut juste se rappeler que chaque cœur peut gérer 2 threads, dit autrement chaque cœur traite deux processus simultanément. Ceci se voit facilement lorsque l'on regarde les processus actifs. Regardez les photos de l'article de Simon Bouby <http://www.leguide3d.com/profiles/blogs/combien-de-ram-ai-je-besoin>.

La question est donc de savoir pour nous si SW10 et 11 utilisent le multi-cœur voire les threads et à quels moments.

Il y a un mode de programmation dit « conçu pour le traitement parallèle » qui permet d'exécuter un programme en le parallélisant.

Pour prendre un exemple simple c'est comme dans la cuisine d'un grand chef, un qui prépare les sauces, l'autre les viandes, encore un autre les légumes, le tout se retrouvant dans la même assiette emmenée par le serveur au bon moment. (Voir le film ratatouille qui explique très bien sur le mode ludique, lui, ce qu'est le multithreading).

Rappelons que du temps de la règle à calcul, on donnait déjà à plusieurs ingénieurs les calculs à réaliser en les découpant en sous-ensembles indépendants.

A contrario avec un programme ordinaire, cela revient à mettre un container sur un bateau rapide. Et même si vous avez 3 autres bateaux qui partent en même temps en convoi, le container n'ira pas plus vite pour autant.

SolidWorks est multithread ? Oui et Non !

Avant de répondre à cette question, il ne faut pas perdre de vue qu'en réalité SW intègre d'autres logiciels écrits à l'origine par des tiers et cela même si les interfaces sont unifiées au sein de SW et rendent cette intégration presque transparente. On peut citer au moins les deux fonctionnalités qui entrent dans ce cadre : la simulation et PhotoView360.

Par ailleurs le SW natif utilise-t-il **le traitement parallèle en bon Français** Gaulois ?

Dans la partie SW pur, seules certaines fonctions utilisent cette possibilité.

- Ouverture de documents dans SolidWorks 2010, l'interface utilisateur et notamment les actions qui réaffichent la ou les boîtes de dialogues, certains sous programmes routines.
- Donc traitement parallèle, lorsque l'on charge un assemblage ou un dessin, le document affiche immédiatement dans un état en lecture seule. L'ensemble des pièces étant récupéré (chargé) en tâche de fond. Pendant le chargement l'information « en lecture seule » indique que vous pouvez utiliser toutes les fonctions prises en charge pour la visualisation (zoom, rotation, etc...), mais il n'est pas possible de passer à un autre document ou de commencer à ouvrir un autre document, ni bien sûr de commencer à travailler le document en cours de chargement. Vous pouvez faire tourner le modèle avec la souris pendant le chargement des pièces en arrière-plan.
- Nous avons aussi le module de simulation (test sur la version SW premium), ce qui se comprend vu la quantité de calculs à effectuer (c'est plus simple de saucissonner un traitement batch, que Word ou Excel).
Le calcul parallèle est le mieux adapté pour le maillage en simulation puisque, pour imager je peux découper la pièce en quatre parties et demander à chaque cœur d'en exécuter un quart mais de façon simultanée et cela peut même aider le petit dernier cœur qui est en retard.
- Egalement, dès que nous passons de SW à Photoview, nous sommes aussi en traitement parallèle.

Dès que l'on lance ne serait-ce que la prévisualisation ou le rendu final sous PhotoView nous constatons que le multi-cœur est sollicité et dès que l'on demande le rendu final les quatre ou 6 cœurs sont à 100%.

(cf. La GPU infra).

Pour le photoréalisme chaque cœur peut calculer chacun des petits carrés qui s'affiche à l'écran (les buckets). On le voit très bien lorsque nous lançons le rendu final avec un quatre-cœur cela traite huit buckets en même temps. Et si l'on lance la même pièce sur un autre ordinateur dual-core, il n'y a pas photo si j'ose dire.

(CF infra la GPU)

Remarque intermédiaire sur le traitement parallèle (multithreading)

SW 2010 et 2011 (PTC est dans le même cas) n'utilise donc pas le calcul parallèle pour la partie modélisation CAO.

Il y a deux explications une officielle et l'autre plus technique.

L'explication officielle est que le processus de résolution utilisé pour la modélisation paramétrique est par nature très linéaire et ne peut pas tirer pleinement profit de processeurs multi-cœurs. A titre d'exemple, on ne peut pas faire l'extrusion N° 2 tant que l'extrusion précédente n'est pas finie. De même, si vous faites la cinématique d'un piston bielle manivelle, tout est synchrone donc pas de parallélisme.

L'explication plus technique est aussi à rechercher du côté du code.

Des programmeurs ayant travaillé à la conversion de programme pour Apple indiquent que c'est extrêmement difficile, car il faut procéder à une analyse complète du code pour identifier ce qui peut être exécuté en parallèle. Les moulinettes de conversion classiques ne peuvent être utilisées.

La seconde explication est que SW est constitué des quelques millions de lignes de code, écrites depuis de nombreuses années par des dizaines de programmeurs ayant chacun leur propre habitude et appelant une multitude de DLL encapsulées que plus personne n'ose toucher, vu la complexité.

D'aucuns pensent que le multithreading est une tendance de fond et que le premier éditeur qui l'adoptera entrainera ses concurrents sur la même voie, tout comme le direct-modeling, il y a 3 ans. A moins que SW ne change totalement de paradigme mais c'est une autre histoire.

Toujours avec le même assemblage utilisé pendant le mini-test réalisé pour le forum Avenao, la vitesse de chargement pour le quatre-cœurs est de xx secondes et de xx secondes pour le dual-core. Toutefois les disques-dur abritant les programmes, du test n'ayant pas du tout la même vitesse, cette indication n'est pas totalement exacte. (Cf, l'annexe)

Le fonctionnement de la CPU

Dans la CPU il y a deux éléments importants : la quantité de mémoire et le processeur.

Comme son nom ne l'indique pas toujours la RAM n'est que la mémoire qui stocke et déstocke de l'information, allant vers et venant du processeur. Dans les faits, une fois le logiciel chargé avec un assemblage de « N » milliers de pièces, la mémoire n'est utilisée qu'à 20%.

Dans le mini-test nous n'avons pas pu déterminer si SW libérait correctement la mémoire (un autre grand éditeur que nous ne citerons pas, est une catastrophe de ce point de vue-là, pour sa version 2010 et antérieures.). Pour SW s'il ne la libère pas, cela vaut la peine d'avoir un peu plus de mémoire. Il faut toujours regarder combien de mémoire on utilise réellement, car c'est souvent la façon de concevoir qui est en cause, plus que le hardware.

La quantité de mémoire.

- Mesure de la RAM utilisée avant le lancement de SW.
3 Giga et 4,5 Giga pour SW 2011 et 5,2 Giga si SW 2010 est lancé simultanément.
- Chargement du même ensemble qui fait N Mo sur les deux sessions de SW une 2010 et une 2011.

Bien sûr il faut un minimum de mémoire pour éviter des accès disques.

Mais contrairement à ce que l'on croit ce n'est pas parce que votre ordinateur fait des accès disques qu'il vous manque de la mémoire. En effet SW ne charge pas tous les programmes dont il pourrait avoir besoin, mais seulement ceux constituant le cœur des fonctions. Les autres seront appelés au fur et à mesure.

Dans ce cas vous constaterez qu'il fait des accès disque pour charger ces bouts de programmes ou diverses DLL. On voit l'intérêt de faire du traitement parallèle pour le chargement (cf supra le traitement parallèle.) et aussi d'avoir un disque dur très très rapide pour les programmes (disque C :).

On peut se poser la question de savoir à quoi cela sert au-delà de 6 méga de RAM?

A noter qu'avoir beaucoup de mémoire et un processeur lent ne sert pas à grand-chose puisqu'elle ne sera pas utilisée lors des calculs.

Pour des compléments sur la quantité de RAM se référer à l'article de Simon Bouby <http://www.leguide3d.com/profiles/blogs/combien-de-ram-ai-je-besoin>.

Windows

Nous n'évoquerons ici que Windows 7 (W7) en 64 bits ou 32 bits. Quel est le rôle de l'operating système (OS) ?

Cette petite précision est juste apportée car vous entendrez dire que W7 est multitâches et qui plus est un multitâches préemptif. Pour faire très simple il faut se dire qu'un programme informatique utilise l'OS pour pouvoir travailler, mais le programme informatique une fois lancé fait ce qu'il veut et déroule sa succession d'ordre selon ce qui lui est demandé, typiquement SolidWorks.

Par exemple on peut écouter de la musique, tout en parlant sur Skype en faisant une démonstration de SolidWorks, pendant qu'une autre version de SolidWorks exécute en arrière-plan un rendu final ou une simulation, le tout pendant qu'un autre programme informatique fait un traitement différé (batch)

Pour reprendre un exemple imagé

Dans les orchestres symphoniques ou dans une harmonie municipale il y a les pupitres avec les trompettes d'un côté, les violons de l'autre, les violoncelles, les hautbois, etc. Dans chaque pupitre vous avez le premier trompette qui n'a pas la même partition que le second et troisième trompette, il en va de même pour chaque pupitre.

Si chacun démarre et s'arrête quand il veut c'est la cacophonie précédant le concert.

C'est alors qu'arrive l'OS (pas celui qui enchaîne chez Renault) mais le master, le Chef d'orchestre au grand cœur.

Un coup de baguette et Hop ! Tout démarre et s'arrête au bon moment, fait une pause, reprend, accélère ralentit, pianissimo, forte. C'est aussi lui qui attribue les ressources du processeur : plusieurs cœurs en vivace pour celui-ci, un seul cœur pour celui-là en andante.

Il est multitâche avec son conducteur qui lit les partitions de tous les pupitres. Il agit en tant qu'ordonnateur des grandes œuvres mais serait bien à la peine pour jouer de la harpe, de la viole de gambe ou du rendering à la façon de SolidWorks.

A chacun son métier et son talent.

Fonctionnement de la GPU (Graphic Process Unit).

En première approche il convient de savoir quand celle-ci est sollicitée ou non.

Cela permet d'éviter une course à la puissance pour les usages courants et même avancés. Cela évite aussi d'imputer des ralentissements ou des défauts d'affichage à la GPU alors qu'elle n'est pas en cause dans le fonctionnement.

De plus comme cela m'a été indiqué par un technicien-produit Avenao, la carte graphique ne s'occupe uniquement de ce qui se voit donc si la majorité des pièces sont cachées cela ne la sollicite pas beaucoup.

Les différents modes d'affichage

Une première approche est de comprendre les différents modes d'affichage car nous verrons que selon le mode utilisé la réponse n'est pas la même.

On discerne :

- OpenGL
- RealView
- Le rendu réel avec :
 - o L'aperçu intégré
 - o La fenêtre d'aperçu
 - o Le rendu final.

Affichage standard.

C'est le cas la plupart du temps en conception de base

Avec des ombrages sommaires qui sont réalisés par OpenGL.

Il ne nécessite pas de matériel spécial et peut être fait par émulation logiciel.

Toutes les cartes graphiques modernes, quel qu'en soit le coût, et la puissance même modeste ont des capacités OpenGL.

Les calculs pour OpenGL sont normalement effectués sur l'unité de processeur graphique (GPU) de la carte graphique, mais dans les faits la CPU agit aussi

Vous remarquerez qu'avec cet affichage standard l'ombre portée reste toujours à la même place. Dit autrement le sol bouge en même temps que la pièce.

RealView.

C'est le second mode d'affichage avec cette fois une ombre projetée dynamique car l'ombre se déplace en fonction des mouvements de la pièce.

N'oublions pas que les lumières sont fixes dans les deux modes d'affichage.

En mode RealView vous avez des ombres sur les pièces entre elles. Ombres d'une partie de la pièce A sur une autre partie de la pièce A ou entre deux pièces ou plus d'un assemblage.

Vous avez également des reflets mais assez limités en termes d'effets.

C'est un peu plus comme dans la vraie vie pas mais tout à fait ;-)

Cela permet de faire des présentations sympas à peu de frais.

En mode RealView pour pouvez commencer à jouer sur les lumières, etc... mais attention si vous allez plus loin dans le rendu réaliste, vous aurez des surprises.

le rendu réaliste (réel)

C'est le troisième mode d'affichage et le fameux rendering avec tout son cortège de sophistications qui permettent d'avoir ces très belles photos qui nous font rêver.

Dans les faits nous trouvons

- o L'aperçu intégré
- o La fenêtre d'aperçu
- o Le rendu final.

Dans ce troisième mode d'affichage on peut l'afficher soit dans SolidWorks en lieu et place de l'affichage standard OpenGL (pauvre et sommaire).

C'est ce que l'on appelle l'aperçu intégré mais c'est très gourmand en ressources d'affichage. Faire une rotation sur un ensemble est pratiquement impossible.

De plus les couleurs ombres, etc, ne sont pas les mêmes que dans la fenêtre d'aperçu ou pendant le rendu final donc surprise à l'arrivée.

Soit l'affichage se fait dans une fenêtre séparée : c'est la fenêtre d'aperçu tout simplement.

Quel intérêt me direz-vous en mode conception ? A mon avis aucun.

Mais pour préparer les scènes, ou des animations la fenêtre d'aperçu a de l'intérêt dans la mesure où elle est très proche du rendu final. Cela permet de travailler les caméras, les ombres, éclairages, etc... L'affichage est également dynamique, alors que le rendu est totalement statique.

Nota :

Il y a une petite astuce qui consiste à paramétrer cette fenêtre en 11 x15 par exemple au lieu de 1920 x 1080. Cela permet de faire tourner la pièce lentement avec pratiquement le rendu final. Très utile pour préparer ses plans caméra ou les éclairages.

Il est possible d'avoir simultanément RealView en même temps que la fenêtre d'aperçu, mais ce n'est pas d'un grand intérêt, me semble-t-il.

Le rendu : sous-entendu le rendu final c'est une seule image qui est générée en mode photo.

Il est possible de générer un film pour les mouvements mais chaque image sera traitée de toute façon une par une.

Le traitement est beaucoup plus poussé puisque tout ce qui est ombre et reflet des objets entre eux, de la scène, etc., sont calculés.

C'est sûrement pendant cette phase que l'intérêt du multithread prend tout son sens. La carte graphique n'est pas du tout utilisée, contrairement à ce que l'on pourrait penser. Donc pas besoin d'une carte très haut de gamme pour le rendering, nous n'avons jamais réussi à faire dépasser les 42% d'activité lors du test.

Pour l'opération de rendu réaliste, la carte a un encéphalogramme plat.

Avant que votre propre encéphale ne vous fasse souffrir, il me paraît raisonnable d'en rester là.

Conclusion

Le multi-cœur permet d'aimer encore plus ! (SolidWorks bien sûr)

Cores dialement

Pierre Gartner

ANNEXE

Les configurations utilisées ne sont pas des références. Elles permettent de comparer les résultats sur deux micro-ordinateurs, toutes choses égales par ailleurs. Le but est mieux comprendre le fonctionnement de SolidWorks et non pas de pinailler sur les vertus de l'overclocking selon les processeurs les jours de vent de noroît dans le bas pays bigouden 😊

SolidWorks 2010 et 2011 en 64 bits

Configuration PC portable Samsung acheté sur étagère. (boîtier jamais ouvert)

OS : Windows 7 64bits

CPU : Intelmobile core 2 duo T6400 à 2GHz

2 threads par core

Mémoire 4096 MBytes

DRAM fréquence 399.0 MHz

GPU ATI Mobility Radeon HD 4650

Taille 1024 MBytes

GPU core clock 220 MHz

GPU memory clock 249.8 MHz

Machine de bureau configurée à la demande

Carte mère gygabyte

2 disques durs :

Un disque 10000 tr/mn pour l'OS et les programmes (facilitent le chargement des modules et des DLL en cours de travail

Un disque dur 1 tera, miroré à chaud

GPU : NVidia quadro 3800 FX, gestion du multi-écran.

2 écrans Viewsonic LCD plein écran affichage 69 cm / 27 pouces

A noter

Nous constatons lors des tests que SW, pour les parties de programme qui n'utilisent pas le traitement parallèle, n'utilise qu'un seul thread sur les 4 disponibles avec l'ordinateur portable et toujours un seul pour les quatre cœurs disposant de 8 threads, idem pour un six cœurs.

De même pour la GPU il faut relire ce que dit un fabricant de cartes à la lumière de ce qui est dit dans le document principal (cf.supra)

Pour la GPU nous voyons ici <http://www.nvidia.fr/object/bfp-solidworks-2011-fr.html> les informations suivantes.

OPTIMISATION DE LA QUALITÉ GRAPHIQUE DANS SOLIDWORKS

Contrairement aux cartes graphiques grand public, les solutions professionnelles Quadro intègrent des capacités RealView et des fonctions d'anticrénelage pleine scène (FSAA). Avec RealView et Quadro, vos modèles 3D prennent vie et interagissent avec leur environnement. Créez des scènes et des apparences uniques puis changez les matériaux, les textures et les environnements afin de donner un aspect réaliste à tous vos modèles. Grâce aux fonctions FSAA, supprimez les effets d'escalier et lissez les contours géométriques pour obtenir des modèles toujours plus réalistes. La puissance des solutions Quadro vous offre un rendu graphique ultra-réaliste de la plus haute qualité. (Source Nvidia)



Mode 3D standard (Realview, FSAA désactivé)

L'image ne contient pas de reflets ou de textures complexes. Les effets d'escalier sont visibles.



Mode 3D avancé (Realview, FSAA désactivé)

Modèle plus réaliste et détaillé. Les ombres, les reflets et les textures s'approchent d'un rendu photoréaliste. Les contours géométriques sont plus lisses.