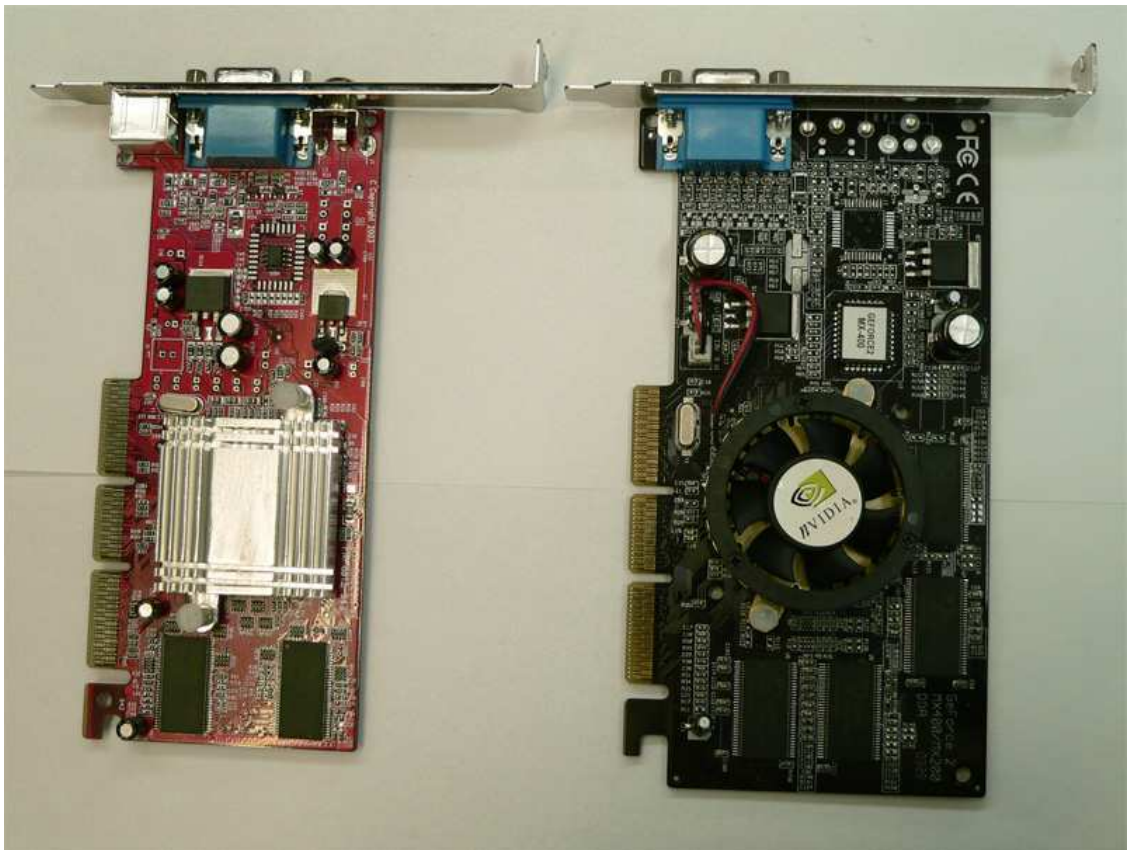


Les cartes graphiques



Abdellah TOUIJRAT

2^{ème} Informatique
2006 – 2007

HELHO – Département Technique

Graduat Don Bosco
Rue Frinoise 12
7500 Tournai

I. INTRODUCTION

Tout ordinateur est doté d'une carte graphique qui assure le passage de l'information entre le système et l'écran. Du plus rudimentaire au plus sophistiqué, ces contrôleurs ont un rôle pratique évident : ils assurent le plus grand confort de l'utilisateur, ou, à tout le moins, tentent de le faire. Pour y arriver, la carte graphique doit réguler au mieux les quatre paramètres majeurs de l'affichage, à savoir : la rapidité d'affichage, la définition de l'écran, la quantité de couleurs affichées, la qualité de l'affichage.

Ce travail va vous permettre de mieux comprendre le fonctionnement d'une carte graphique, ainsi que sont évolution au fil du temps. Avec ses connaissances supplémentaires vous serez à même de choisir la carte graphique la plus appropriée à vos besoins, car l'univers des cartes graphiques, monde sans pitié, est sujet à de profonds et rapides renversements de tendances.

De plus, les ordinateurs sont vendus avec des numéros ; plus le numéro est élevé, mieux c'est. DirectX 9 sonne mieux que DirectX 8 et 256 Mo de mémoire vidéo sont à l'évidence préférables à 128 Mo. Beaucoup de consommateurs moins bien informés font beaucoup trop confiance à cette philosophie du « plus le chiffre est élevé, meilleur c'est ». Nous espérons que ce travail vous permettra d'être plus critique à l'avenir.

La carte graphique (écran) est la carte d'extension qui transforme un flux de données digitales en un signal analogique compréhensible par un écran. Les cartes graphiques Hercules furent les premières cartes dédiées au PC, elles étaient en noir et blanc (ou plutôt selon le moniteur ambré). La couleur est réellement apparue avec les premiers 286. A cette époque, la norme était le CGA ou EGA, le moniteur devait être compatible avec la carte écran. Utilisant une résolution et un nombre de couleurs restreint, cette norme fut remplacée par la norme VGA. La norme VGA permettait d'afficher une résolution de 640 * 480 points en 16 couleurs. Une carte graphique VGA inclut 256 k de mémoire.

Comme chaque constructeur souhaite toujours améliorer, chacun a sorti des cartes graphiques S-VGA (pour Super VGA). Ces cartes ne répondent pas à une norme, chaque constructeur développe un pilote (un programme) qui permet au système d'exploitation de gérer ses possibilités. Chaque pilote est spécifique au modèle de carte graphique mais aussi au système d'exploitation. Le mode VGA est reconnu par tous les systèmes d'exploitation actuels. Pour passer en mode supérieur, un pilote-driver (le programme spécifique au matériel) doit être installé.

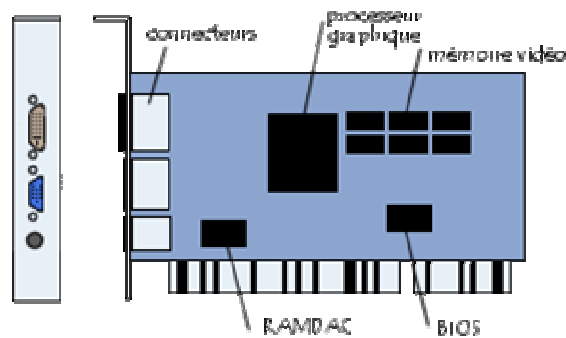
Attention, quelques pilotes Win95 **ne sont pas compatibles** avec Win98. Néanmoins, comme le nombre de constructeurs de circuits intégrés spécialisés carte écran est limité, généralement, un pilote S3 DX2 (un exemple de circuit d'affichage) est compatible avec toutes les cartes incluant le même circuit, quel que soit le constructeur.

Une zone mémoire est réservée dans la mémoire haute depuis les premiers 8088. Elle est actuellement largement inférieure à celles utilisées par une carte graphique (jusque 256 MB) et peut être récupérée par les commandes DOS de mémoires hautes. Le processeur graphique garde les zones non modifiées dans sa propre mémoire. Ceci permet de ne pas trop utiliser le processeur.

II. Caractéristiques d'une carte graphique (écran).

Une carte écran se caractérise par:

- Bus utilisé (ISA, PCI, AGP et PCI express)
- Le processeur graphique
- Mémoire
- Convertisseur digital – analogique (RAMDAC)
- Taux de rafraîchissement maximum.
- Les fonctions d'une carte graphique (2D, Possibilités 3D)
- BIOS Vidéo
- Connecteur de sortie



Quelques unes de ces caractéristiques sont identiques à celle d'un système à microprocesseur: processeur, mémoire et Bios. En effet, le sous système vidéo est un système à microprocesseur tout à fait conforme dont le seul but est de traiter des informations digitales provenant du PC pour les transférer vers le convertisseur.

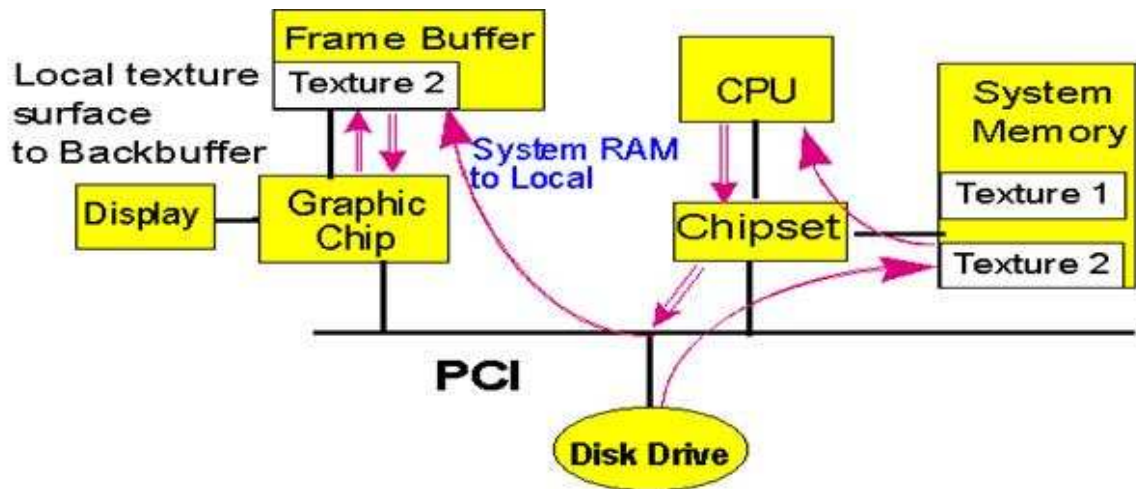
iii. Le bus utilisé.

Pour communiquer avec l'ordinateur, la carte graphique reçoit et transmet les informations par un « bus ». L'installation d'une carte graphique dépend forcément du bus utilisé. Les performances en dépendent. Rien ne sert de rouler en Ferrari sur des chemins de terre.

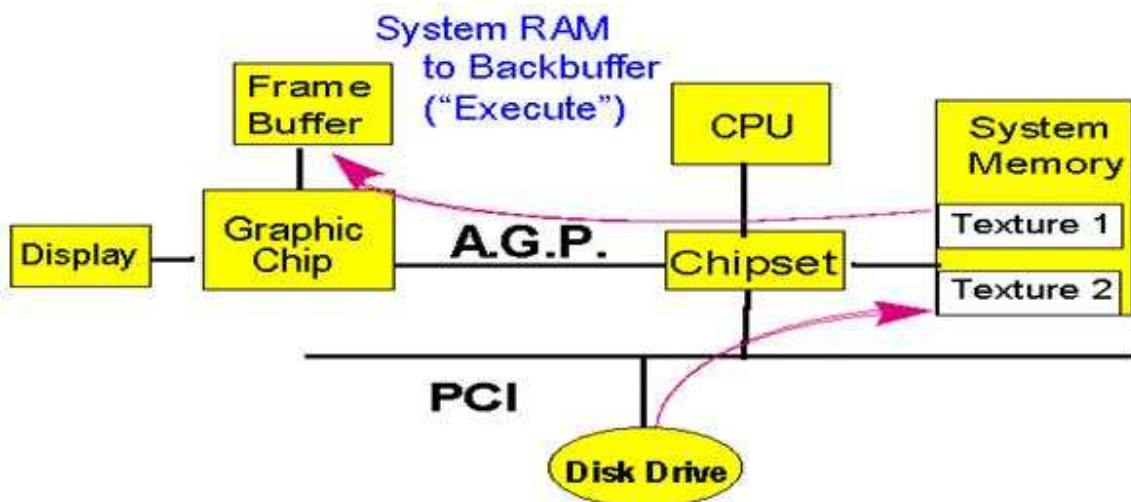
Dans un premier temps le bus ISA été utilisé mais qui était limité à 16 MB/s. Durant plusieurs années la carte utilisait le bus PCI à 132 MB/s mais il a été remplacé par le bus AGP qui a fait son apparition il y a déjà quelques années, l'AGP 2X à 528 MB/s et l'AGP 4X culmine à plus de 1,7GB/s (en théorie).

Comparons les performances du bus PCI et AGP en analysant le parcours emprunté par une information à traiter :

dans un système PCI, la texture passe du disque dur à la mémoire centrale, transite via le chipset sous le contrôle du processeur puis repart par le bus PCI vers la mémoire tampon (buffer) de la carte graphique. Dans le cas d'une carte graphique PCI bus master, le transfert peut se faire sans intervention du processeur. Finalement, elle arrive au circuit graphique où elle est plaquée sur son polygone avant d'être affichée à l'écran. Cette longue procédure vient du fait que les cartes PCI ne savent texturer que depuis leur propre mémoire.

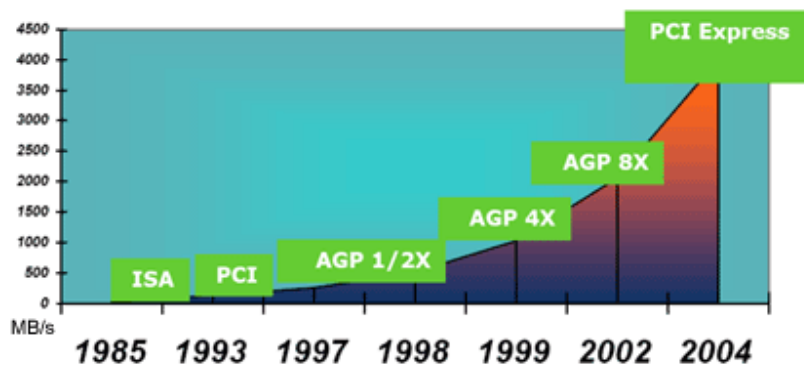


Les cartes AGP sont bien plus performantes, il suffit de faire transiter la texture du disque dur vers la mémoire centrale. Le chipset graphique AGP peut alors directement effectuer des opérations en mémoire centrale.

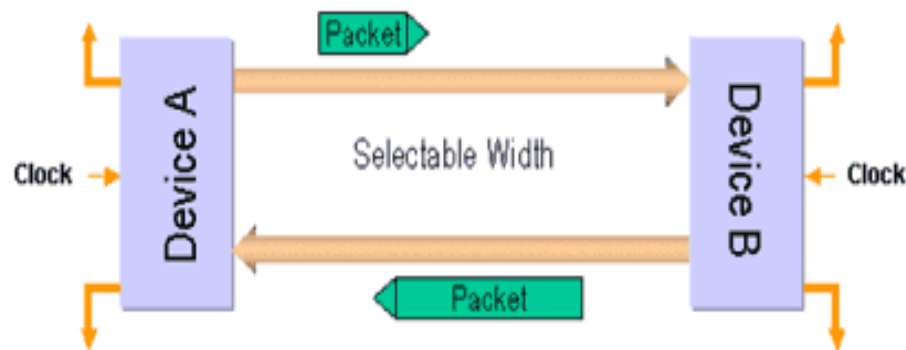


On comprend mieux pourquoi le bus AGP est utilisé par la quasi totalité des cartes graphiques actuelles. Il permet une liaison directe entre la carte graphique, le processeur et la mémoire vive et offre une bande passante bien plus élevée que le bus PCI. Le gain de performances est énorme et certaines cartes AGP ont des performances près de 20 fois plus importantes que le même modèle en version PCI.

La plupart des cartes graphiques sont encore format AGP, qui sert sur la majorité des ordinateurs. Le nouveau bus, prévu pour remplacer à la fois le PCI et l'AGP à moyen terme, s'appelle le **PCI Express**, souvent abrégé en PCIE ou PCIX. Cela dit, malgré le nom similaire, il n'a rien à voir avec le bus PCI-X que l'on trouve sur beaucoup de cartes pour station de travail et serveurs. Mais on en trouve déjà en PCI Express, ce nouveau bus haute vitesse appelé à remplacer l'interface AGP ; elles ne peuvent être installées que sur des PC de dernière génération, équipés du connecteur idoine. Certains circuits graphiques sont utilisés dans les deux formats, ils délivrent les mêmes performances. Les cartes PCI, uniquement utiles sur les PC très anciens, sont de plus en plus rares.



. Un bus PCI Express permet de transférer des données dans les deux sens en même temps.

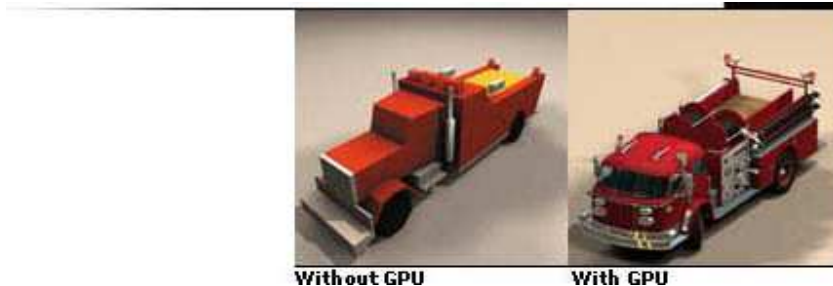


L'infrastructure de l'architecture PCI Express diffère totalement de celle du bus PCI « classique ». D'une part, le PCI Express utilise un schéma de transfert de données en série, qui autorisent des fréquences d'horloge très élevées. Cela, ajouté au fait que plusieurs canaux (ou voie) PCI Express peuvent être groupés sur chaque slot d'extension, laisse de la marge pour plusieurs paliers d'extension comme 8x et 12x. Enfin, le même débit est possible dans les deux sens simultanément.

Pour l'instant, il semble que le slot d'extension standard sur les futures cartes mères sera le PCI Express 1x. Dans ce cas, le « 1x » signifie que le slot a une voie PCI Express, soit une bande passante de 250 Mo/s (500 Mo/s si l'on compte la bande passante pour les deux sens, c'est à dire le full-duplex), ce qui est presque deux fois plus rapide que le PCI. De plus, les différents composants n'auront plus à se battre pour avoir un bout de bande passante, car chaque slot aura l'intégralité des 250 Mo pour lui tout seul.

Les cartes graphiques utiliseront le slot PCI Express 16x. Vous avez bien lu, cela veut dire 16 voies, soit une bande passante maximum de 4 Go/s , ou 8 Go/s si l'on compte les 4 Go/s dans les deux sens. Cela dit, les chiffres donnés pour le full-duplex doivent être pris comme des arguments marketing et ne sont pas vitaux pour l'utilisateur dans la mesure où actuellement seul le débit dans un sens est important pour la vitesse graphique.

IV. Le processeur graphique (GPU)



Sur une carte vidéo, le processeur est aussi appelé chipset et peut être considéré comme le coeur de la carte. Il est composé d'une multitude de transistors (des millions voir même une dizaine de millions) servant aussi bien à la partie traitement 2D et 3D que la partie vidéo. De la qualité de celui-ci, dépendent les performances de la carte. Sous ce terme se cache l'ensemble des processeurs de cartes graphiques qui vont effectuer les calculs nécessaires à la création de l'image qui sera affichée. En fait, tout comme sur la carte mère de l'ordinateur, le chipset graphique contrôle chacune des fonctions essentielles de la carte vidéo.



On détermine le niveau de qualité d'un Chipset graphique en regardant sa largeur maximum de bande passante. La largeur de bande est tout simplement l'indication de la vitesse de circulation des données qui y transitent. L'ordre de grandeur de cette bande passante est exprimé par le nombre de Gigaoctets traités en une seconde. De nos jours, deux types de Chipset graphique sont utilisés : les Chipsets dits de 256-bits et les chipsets 128-bits. Les chipsets 128-bits étant les plus anciens, il s'agit du coeur de la carte graphique. C'est-à-dire que c'est au niveau de ce composant que sont traités tous les calculs. Ces calculs étant de plus en plus complexes, des fonctions spécifiques sont implantées dans

le chipset. On parle même d'algorithmes au lieu de fonctions et d'accélérateur graphique au lieu de chipset.

Un processeur graphique peut alors se caractériser par sa capacité à effectuer des calculs plus ou moins complexes en un minimum de temps. Lors de l'achat d'une carte vidéo, nous pouvons parfois lire sur la boîte "moteur de calcul capable de reproduire jusqu'à 6 millions de triangles par seconde". Ceci peut être un indicateur relativement intéressant concernant la puissance du processeur graphique. Plus ce nombre est grand, plus la carte est puissante.

Comme pour le processeur central de l'ordinateur, le processeur graphique est cadencé à une certaine vitesse. Cette vitesse s'exprime par une fréquence en MHz. Plus cette fréquence sera élevée, plus les informations seront traitées rapidement et plus l'affichage final sera rapide. Cependant il faut noter que faire fonctionner des processeurs graphiques à une fréquence élevée implique un échauffement certain de ces derniers.

Parfois, certains chipsets peuvent monter en température jusqu'à 100°C. Afin de préserver celui-ci, il n'est pas rare de rencontrer un dissipateur thermique placé sur l'accélérateur graphique. Parfois, un simple dissipateur ne suffit pas et dans ce cas, on le remplace par un véritable ventilateur afin de limiter l'augmentation de la température et de garder celle-ci constante.

V. Les méthodes de refroidissement "traditionnelles"

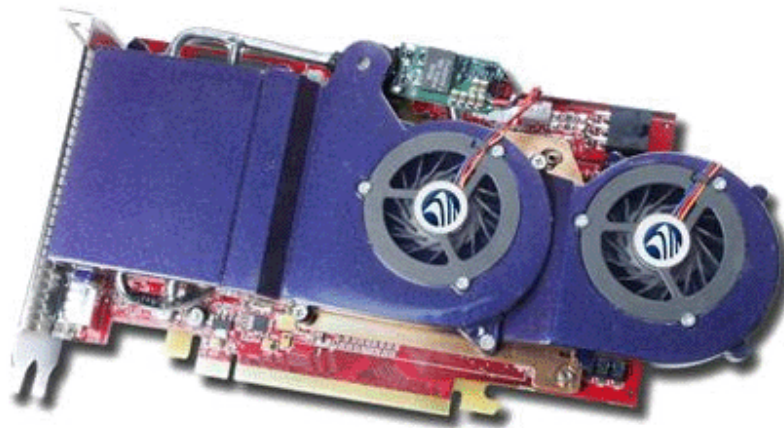
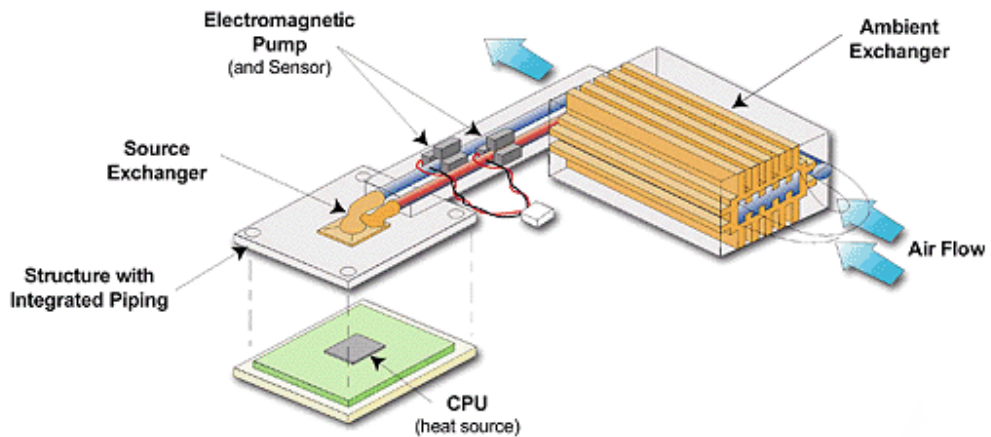
le refroidissement à air passif : un simple dissipateur thermique, ou radiateur, appliqué sur la pièce à refroidir. Ce système était utilisé sur pratiquement tout les processeurs pour PC du i386 aux premières générations de Pentium. Il est encore utilisé pour refroidir les modèles les moins exigeants de processeur graphique et de Northbridge ainsi que quelques processeurs étudiés pour avoir une dissipation thermique moindre.

- le refroidissement à air actif : on adjoint un ventilateur au radiateur pour forcer le flux d'air à mieux le refroidir (l'ensemble ventilateur-radiateur est parfois appelé ventirad). Ce procédé constitue le standard en ce qui concerne le refroidissement des CPU de plus de 100 Mhz, il est également utilisé sur certaines carte graphiques haut de gamme aux processeurs graphiques ayant de gros besoins de dissipation thermique ainsi que dans la plupart des blocs d'alimentation.
- le procédé du caloduc reprend le concept du refroidissement à air actif avec le couple radiateur/ventilateur, à la nuance près qu'au lieu d'employer un simple dissipateur thermique en aluminium et/ou cuivre il se présente sous la forme d'une enceinte hermétique qui renferme un liquide en équilibre avec sa phase vapeur, en absence de tout autre gaz. Ce procédé très récent dans le domaine du refroidissement donne des résultats approchant ceux du refroidissement à eau et séduit ceux qui sont rebutés par la complexité et le risque de ce dernier.

les deux premières cartes graphiques utilisant un refroidissement par un liquide métallique, technologie mise au point par Nanocooler.

Il s'agit en fait d'un refroidissement liquide de taille réduite (watercooling) mais avec un liquide métallique (le gallium, comparable au mercure pour vous donner une idée), l'avantage de ce caloporteur est que sa conductivité thermique est plus élevée mais que celle de l'eau. La circulation du liquide est assurée par une pompe électromagnétique (sans éléments mécaniques).

Sapphire annonce 10° de moins qu'un système Artic Cooling et une nuisance sonore réduite (de seulement 25% par rapport au système de ventilation ATI classique) sur les 2 premières cartes Blizzard Radeon X850 XT et X850 XT Platinum Edition. L'intégration de ce système sur des cartes moins puissantes devraient toutefois permettre de les refroidir de manière totalement passive (et silencieuse) comparé aux cartes refroidies par de simples caloducs .



vi. La mémoire.

La mémoire de la carte graphique permet d'augmenter la résolution tout en gardant un nombre de couleurs respectables. Par exemple, un écran de $1024 * 768$ pixels en 2 couleurs (noir et blanc) nécessite $786.432 \text{ pixels} * 1 \text{ bit} / 8$ (pour codage en octets)=98 K. Pour 4 couleurs (nb. Octets =2), 196,6K et en 16 couleurs (4 bits):384 K. Ceci donnerait pour $1600 * 1200$ en 16 millions de couleurs (14 bits)=30.000 MB.

Résolution	Nombre de couleurs	Mémoire
640 * 480	256	512K
	32 K	1 MB
	64 K	1 MB
	16 Millions (24 bits)	1 MB
800 * 600	16	512 K
	256	512K
	32 K	1 MB
	64 K	1 MB
	16 M	2 MB
1024 * 768	16	512K
	256	1 MB
	32 K	2 MB
	64 K	2 MB
	16 M	4 MB
1280 * 1024	16	1 MB
	256	2 MB
1600 * 1200	256	4 MB
	64 K	4 MB

Le volume de **RAM vidéo** est probablement l'élément le plus surestimé d'une carte graphique. Les acheteurs mal informés ont tendance à considérer le volume de RAM pour différencier une carte d'une autre, alors que le volume de RAM a un très faible impact sur la vitesse par rapport à d'autres considérations comme la fréquence d'horloge et l'interface mémoire.

En règle générale, une carte avec 128 Mo de RAM tournera presque aussi rapidement qu'une carte avec 256 Mo dans la plupart des situations. Il y a des situations où avoir plus de RAM est avantageux

pour la vitesse, mais il est important de garder à l'esprit qu'avoir plus de RAM ne se traduit pas automatiquement par une vitesse plus élevée.

Là où le volume de RAM se révèle important c'est dans les environnements avec des textures haute résolution. Les développeurs de jeux créent souvent des environnements avec des textures multiples pour leurs jeux et, plus le volume de RAM de la carte graphique est important, plus l'on peut utiliser des résolutions de texture élevées. Les textures haute résolution sont plus nettes sur les jeux. Cela dit, le bon sens commande de choisir une carte avec le plus de RAM possible, mais uniquement si toutes les autres considérations sont égales. Malgré tout, le bus mémoire et la vitesse d'horloge de la mémoire influent bien plus sur la vitesse que le volume total de mémoire présent sur la carte.

Dans le cas des cartes 3D, une large partie de la mémoire est utilisée pour enregistrer des textures. Une carte 3D 32 MB ne permet donc pas d'utiliser des résolutions très élevées. De toute façon, aucun écran actuel n'est capable d'afficher les résolutions maximum des cartes écrans. Pour rappel, l'œil humain est limité à 16 Millions de couleurs.

Le processeur est relié à la mémoire à travers un large bus compris entre 64 et 128 bits.

Les types de mémoires utilisées ont également leur importance.

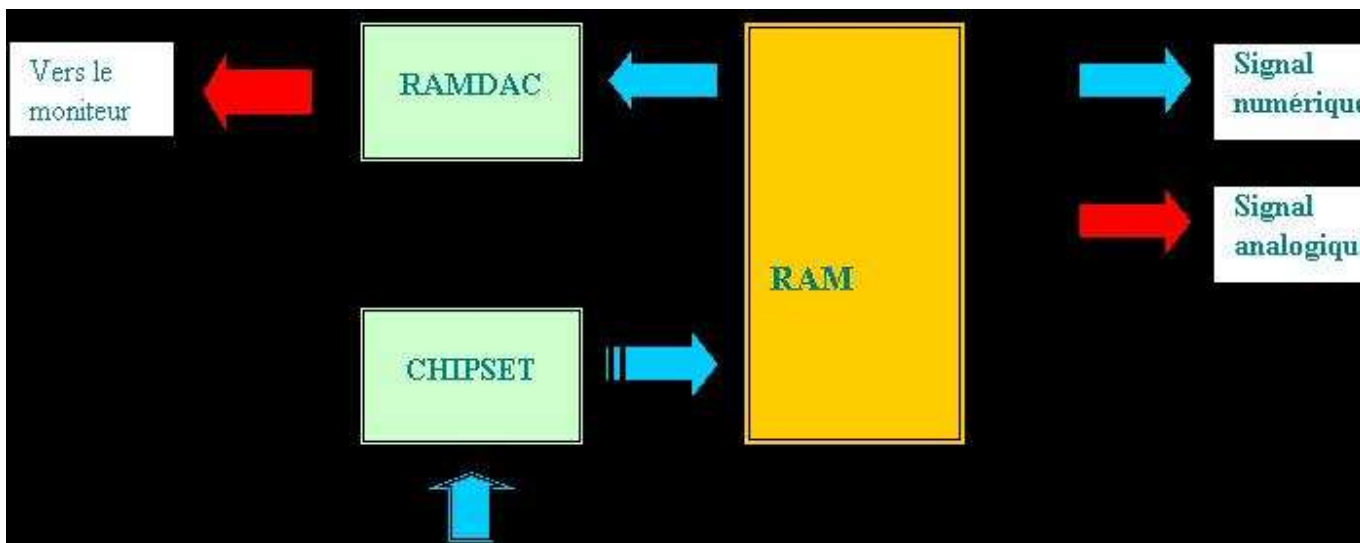
Il y a deux catégories de mémoire : SDR(débit unique) et **DDR** (double débit), qui transfère deux fois plus de données par cycle d'horloge. la mémoire à débit unique est depuis longtemps obsolète sur les cartes graphiques. Comme la **mémoire DDR** fait deux fois plus de travail que la SDR, il est important de se souvenir que la fréquence d'horloge de toutes les mémoires DDR sont généralement indiquées comme du double de la fréquence d'horloge réelle. On parle par exemple de mémoire DDR 1000MHz (c'est à dire 1000MHz efficaces) alors que la fréquence d'horloge réelle est de 500 MHz.

Pour cette raison, beaucoup de gens sont surpris quand leur carte graphique avec de la mémoire DDR à 1200MHz est indiquée comme tournant à 600MHz. Il n'y a pas de quoi s'alarmer, c'est seulement la façon dont est indiquée la fréquence de la mémoire DDR. Les mémoires DDR2 et GDDR3 fonctionnent sur le même principe, avec une fréquence d'horloge doublée. La différence entre les mémoires DDR, DDR2 et GDDR3 tient uniquement à la technologie de fabrication. La DDR2 peut généralement tourner à des fréquences plus élevées que la DDR, et la DDR3 à des fréquences plus élevées que la DDR2.

vii. Le convertisseur digital – analogique(RAMDAC)

Le rôle du Ramdac est essentiel pour l'affichage. En effet, les informations qui transite dans l'ordinateur sont codées sous forme binaire, autrement dit, de façon numérique. Or, les moniteurs à tube cathodique sont incapables de comprendre les données binaires. Ils ne peuvent afficher que des données analogiques. C'est pourquoi il faut qu'entre la carte vidéo (signaux numériques) et le moniteur (signaux analogiques) une conversion des données soit faite : c'est exactement ici qu'intervient le Ramdac. C'est lui qui est chargé de cette transformation du signal. Les performances du Ramdac doivent être à la hauteur des possibilités de l'affichage, qu'il s'agisse de la carte vidéo ou du moniteur. Un Ramdac trop faible limitera les performances de l'un ou de l'autre. Plus la fréquence sera élevée, plus l'affichage pourra être effectué à une résolution importante en conservant un taux de rafraîchissement satisfaisant. La plupart des cartes actuelles ont un Ramdac cadencé à 250 MHz ce qui est déjà très suffisant

Dans la figure ci-dessous, nous pouvons mieux comprendre le rôle du Ramdac. On voit bien qu'il reçoit un signal numérique stocké en mémoire puis converti celui-ci en un signal analogique qui est envoyé vers le moniteur.



Le **RAMDAC** est doté d'une mémoire vive destinée à stocker des palettes de conversion qui seront utilisées si l'image affichée, ou le mode d'affichage, n'est pas en 16 millions de couleurs au minimum.

Il faut savoir que le **RAMDAC** n'est pas toujours utilisé. En effet, seules les Prises VGA et S-Vidéo nécessitent un signal analogique : dans le cas de ces deux prises, le **RAMDAC** doit faire la conversion. Dans le cas de la Prise DVI, la conversion n'est pas nécessaire, car DVI envoie directement des signaux numériques vers le moniteur (écran), ce qui entraînera un rendu rapide, donc de meilleure qualité.

VIII. Taux de rafraîchissement maximum.

A l'instar du cinéma ou de la télévision, votre **ordinateur** simule le mouvement sur votre écran en affichant une série d'images différentes. Le taux de rafraîchissement est le nombre de fois où la carte graphique rafraîchit cette **image** en une seconde. Un taux de rafraîchissement de 75 Hz signifie que l'image de l'écran est rafraîchie 75 fois par seconde.

Le taux de rafraîchissement peut poser problème avec un **jeu vidéo** quand l'ordinateur traite les images plus rapidement que l'écran ne les rafraîchit. Par exemple, si l'ordinateur est assez rapide pour traiter 100 images à la seconde et que le taux de rafraîchissement de l'écran n'est que de 75 Hz, il y aura des moments où une image sera calculée et affichée à moitié pendant un rafraîchissement de l'écran. Cela provoque des « déchirures » ou des « effets parasites », ce qui est ennuyeux.

Comme remède à cela, on peut activer V-sync (un raccourci pour synchronisation verticale). Elle limite les images que traite l'ordinateur au taux de rafraîchissement exact de l'écran et évite les effets parasites. Par exemple, quand V-sync est activé le nombre d'images calculées par seconde dans un jeu n'excédera jamais le taux de rafraîchissement. Un taux de rafraîchissement de 75 Hz empêchera l'ordinateur de calculer plus de 75 images par seconde.

Quand vous choisissez les paramètres sur lesquels va s'aligner votre carte graphique (taux de rafraîchissement) vous devez prendre un soin minutieux afin de vous assurer que votre moniteur peut effectivement supporter cette fréquence de taux de rafraîchissement sinon vous risquez de réduire de façon très significative l'espérance de vie de votre moniteur et éventuellement causer des dommages irréparables aux circuits électroniques internes très sensibles de votre moniteur. Vous ne

devez jamais au grand jamais utiliser des taux de rafraîchissements supérieurs à ceux qui sont supportés par votre moniteur au risque de le voir partir en fumée!

IX. Les fonctions d'une carte graphique.

A. La 2D

L'affichage en deux dimensions est le plus courant. C'est l'affichage de base. Il est utilisé par MS-DOS, Windows et par tous les logiciels fonctionnant sous un système d'exploitation. La 2D consiste tout simplement à la représentation d'une scène selon 2 axes de référence (x et y). Cela peut aller d'une simple image à un ensemble d'images nécessitant des calculs assez complexes.

Sous Windows, l'affichage est principalement en 2D (fenêtre, curseur de souris, etc.). De nos jours tous les processeurs graphiques intègrent les primitives de Windows. C'est-à-dire qu'ils sont capable avec un minimum d'instructions de créer une fenêtre bien précise en un minimum de temps. A titre d'information, les premières cartes graphiques ne géraient pas l'affichage 2D sous windows. Les fenêtres et curseurs étaient alors calculés de manière logicielle, ce qui prenait parfois beaucoup de temps. Heureusement, dans les nouveaux chips graphiques, toutes les fonctions 2D de Windows sont implémentées de manière matérielle.

La partie 2D d'une carte graphique peut se résumer à l'ensemble des calculs et du temps d'exécution pour traiter une scène en 2D. Ce traitement peut aller du simple affichage d'une image à la manipulation ou transformation d'images plus complexes.

B. Possibilités 3D

Aujourd'hui, la 3D est le domaine à la mode dans le monde informatique. L'essor actuel de la 3D se manifeste dans de nombreux domaines et en particulier dans les logiciels de jeu. A travers ces jeux, nous sommes amenés à manipuler des objets et des personnages virtuels qui évoluent dans des décors ou des paysages fictifs. Cela ressemble à la réalité, et pourtant chaque image de ces séquences animées est calculée en temps réel par l'ordinateur. Les technologies

utilisées pour créer cette impression de réalisme des décors, des objets, des personnages sont en fait très élaborées. Le PC doit effectuer une masse énorme de calculs pour produire ces images en 3D à une cadence suffisante de 25 images par seconde (fluidité d'image de la télévision). Mais pour soutenir ce rythme, le processeur de l'ordinateur ne suffit pas. Il doit être aidé par un calculateur spécialisé, appelé accélérateur graphique (chipset de la carte graphique).

A l'intérieur de cet accélérateur graphique, plusieurs fonctions 3D ont été implantées. Ces fonctions ont pour but de faciliter les calculs d'images 3D. Afin de comprendre le rôle et surtout la nécessité de ces fonctions, voyons d'où proviennent tous ces calculs.

Le procédé 3D consiste à créer une forme à l'aide de polygones, puis d'appliquer une image (la texture) sur ces polygones, et enfin d'y appliquer des effets, comme la transparence, l'effet brouillard, le lissage des angles, etc. Le but est de rendre une image qui soit la plus proche possible de la réalité, avec un effet de profondeur (la perspective) qui lui donnera l'aspect 3D, de la même façon que nos yeux voient les objets dans la réalité. Il existe plusieurs méthodes de représentation visuelle des objets en 3D.

Premièrement, les personnages, les objets et les décors des jeux 3D sont constitués de points (repérés dans l'espace 3D) reliés entre eux par des segments de droites qui délimitent des polygones. La représentation visuelle la plus simple des maillages polygonaux est de type "fil de fer"

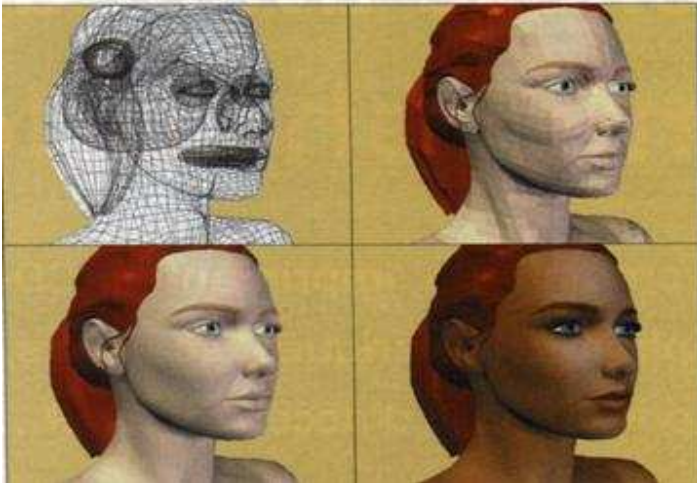
Dans ce mode, seul le squelette de l'objet apparaît et il n'y a pas de notion de surface. Une image 3D est donc à la base qu'un ensemble de triangles formant des polygones. Sur certaines boîtes de cartes graphiques nous pouvons voir le nombre de polygones susceptible d'être calculé par l'accélérateur graphique.

Deuxièmement, le mode de représentation en volume le plus simple

attribue une couleur à chaque polygone et fait disparaître de l'affichage toute les surfaces qui ne sont pas visibles pour l'observateur. Le rendu est alors nettement plus réaliste que le volume fil de fer. Cependant il conserve tout de fois un aspect anguleux, avec des facettes, qui trahit l'origine polygonale du modèle

Troisièmement, le lissage dit de Gouraud vient résoudre le précédent problème. Il fait disparaître les facettes et donne l'impression d'arrondir les angles. A partir des valeurs déterminées pour les sommets de chaque polygone, la couleur de chaque pixel est calculée par interpolation linéaire. Les jonctions entre les polygones deviennent alors invisibles, et les maillages, même grossiers apparaissent lissés.

Enfin, l'application de textures bitmap (images en 2D plaquées sur le volume en 3D) améliore nettement l'aspect réaliste de l'objet représenté. Le nombre de polygones ne change pas, mais la texture apporte des détails et des couleurs. C'est l'emploi de nombreuses textures qui explique la qualité graphique des jeux 3D actuels, car en fin de compte les modèles polygonaux utilisés sont en fait grossier.



Différentes étapes de modélisation d'une image 3D

Comme nous venons de le voir, pour générer une image en 3D, il faut plusieurs étapes. C'est de la dernière étape, application de texture, qu'est née une multitude de fonctions 3D. Le rôle des textures est essentiel dans toute représentation en trois dimensions un tant soit peu réaliste.

De nombreux algorithmes ont été développés pour en tirer meilleur parti. C'est surtout au niveau de cette étape que l'ensemble des calculs seront effectués. Il existe une multitude de techniques pour plaquer une texture. On peut plaquer sur un objet 3D différents types d'informations : couleurs, luminosité, réflexion, reliefs. Dans la pratique, pour obtenir l'effet final, plusieurs textures sont empilées sur un même objet.

La capacité à effectuer plusieurs passes de placage en un cycle horloge est essentielle pour une carte 3D de nouvelle génération. Voici quelques fonctions 3D primordiales supportées par les accélérateurs graphiques récents (à partir de 2000). L'ensemble de ces fonctions ont pour objectif premier de rendre plus réel un objet modélisé en 3D.

a) **Alpha Blending :**

Cette fonction permet de faire varier le degré d'opacité ou de transparence d'un pixel. Il est utilisé pour les effets de brouillard, et sert aussi à simuler des vitres et des liquides.

Un programmeur pourra par exemple représenter un fantôme en affichant un sprite à demi-transparent au dessus du décor du jeu. La GameBoy Advance propose plusieurs mécanismes dédiés à la réalisations d'effets de transparence, mais également de fade-in et de fade-out. Pour calculer la couleur d'un pixel supposé transparent voici la méthode à employer :

$$r = \text{source1.r} * \text{eva} + \text{source2.r} * \text{evb}$$
$$g = \text{source1.g} * \text{eva} + \text{source2.g} * \text{evb}$$

$$b = \text{source1.b} * \text{eva} + \text{source2.b} * \text{evb}$$

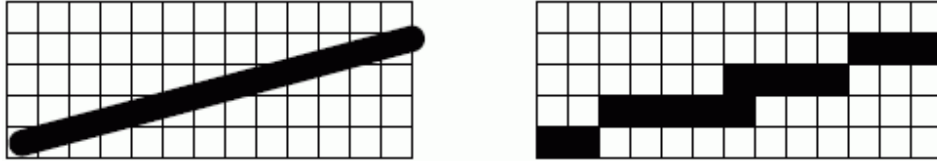
Les structures source1 et source2 correspondent aux pixels des deux sources ayant un rôle dans l'effet de transparence. Une transparence agit entre deux objets, par exemple entre un sprite et un background ou entre deux backgrounds. La formule précédente mélange les couleurs (c'est pour cela que nous utiliserons dans nos programmes la macro BLEND) entre les deux sources et remplace le pixel correspondant à la superposition des deux sources. Une opération d'alpha-blending peut se voir paramétrée sur 16 niveaux de transparence. La valeur eva agit sur la première source tandis que le coefficient evb agit sur la seconde source

b) Anti-aliasing :

Cette fonction permet d'atténuer, sinon de supprimer les effets d'escalier apparaissant sur les bords des objets en 3D. Elle peut être réalisée sur une partie ou sur la totalité de l'image (on parle alors de full Scene anti-aliasing). Cette fonction est extrêmement gourmande en calculs.

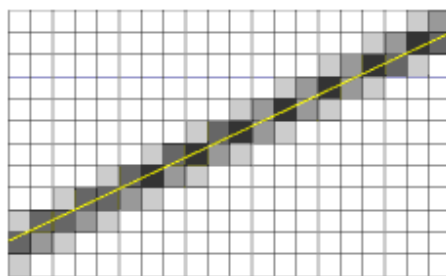
Il existe plusieurs façons de voir l'antialiasing ; une d'entre elles consiste à le considérer comme une méthode permettant de représenter des vecteurs parfaits et continus sur un périphérique d'affichage imparfait et techniquement limité. A la base, les jeux utilisent pour représenter la réalité des formes géométriques ayant comme primitive les triangles. Ceux-ci, en tant que figure géométrique, sont donc lisses et parfaitement continus. Par contre, un moniteur possède une caractéristique essentielle : il affiche les images en utilisant une grille, qui représente en fait la résolution. Celle-ci est limitée à un certain nombre de pixels; dès lors, elle ne peut afficher qu'un nombre fini de nuances au niveau des courbes ou des vecteurs non parallèles avec les bords de l'écran. C'est alors qu'un effet d'escalier apparaît, mettant

clairement en évidence les limitations de l'écran pour se rapprocher d'un rendu réaliste.



Une solution pour diminuer cet effet consiste à augmenter la résolution de l'écran dans les options des jeux. Le seul problème est qu'aujourd'hui, la puissance des cartes graphiques augmente bien plus rapidement que les capacités des écrans, et que rares sont ceux pouvant dépasser le 1600 x 1200, voir le 1920 x 1440 (sans même parler de la fréquence de rafraîchissement, si importante pour éviter la fatigue visuelle). Evidemment, ce problème est amplifié avec les écrans TFT, calibrés pour une seule résolution et qui doivent réaliser une interpolation des pixels (pas toujours réussie, il faut bien l'avouer) pour afficher une résolution supérieure. La résolution de base des TFT est généralement de 1280x1024 pour les 17". Or on constate aujourd'hui que l'œil humain est capable de distinguer les effets d'escaliers jusqu'à une résolution de 4000x4000, voir au-delà.

L'antialiasing est donc plus précisément une technique permettant de diminuer les effets d'escalier sans modifier la résolution de l'écran. Il fut inventé par le Media Lab du Massachusetts Institute of Technology (MIT), et il est à noter qu'il ne se limite pas à l'image puisque dans le domaine du son, l'aliasing est un phénomène qui est éliminé en supprimant les fréquences situées au-delà de la moitié des fréquences de sampling.



Restait à connaître l'impact concret de l'activation de l'antialiasing tel que géré actuellement, sur les jeux. Pour ce faire, nous nous sommes placés devant une arche sur Serious Sam : Second Contact, et nous avons comparé les effets d'escaliers obtenus en montant la résolution d'une part, puis en restant en 1024x768 mais en augmentant l'antialiasing d'autre part. L'antialiasing utilisé était celui des cartes ATI de génération R3X0/R420 (voir détails plus bas).

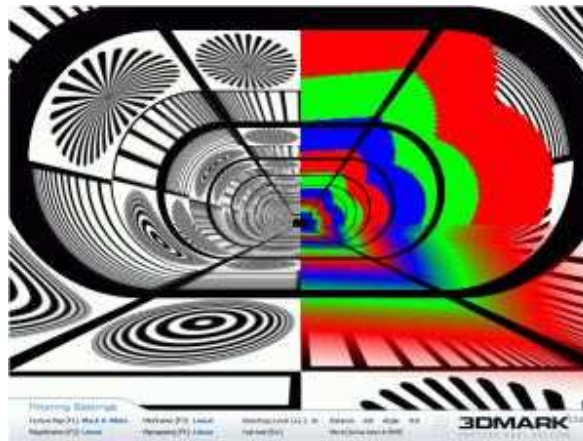
Note : attention à ne pas se focaliser sur l'effet d'escalier visible au niveau du reflet du ciel sur l'arche. En effet, seul l'augmentation de la résolution peut améliorer ceci, l'antialiasing n'intervenant que sur les extrémités de l'arche.

c) **Bilinear Filtering** :

Le filtrage des textures est une opération importante et omniprésente dans les jeux. Dès que la distance de vue dépasse les quelques mètres et qu'une texture se prolonge (typiquement, celle du sol), c'est lui qui intervient afin de faire en sorte que les transitions restent invisibles sur cette texture, plus précisément les mipmaps. Que sont les mipmaps ? Des versions basse résolution d'une texture, qui sont appliquées sur les polygones éloignés du joueur, partant du principe qu'une surface éloignée d'un kilomètre n'est pas perçue avec autant de détails qu'une surface située à un mètre. Les mipmaps correspondent donc en quelque sorte à l'antialiasing des textures ; un antialiasing précalculé ce qui permet d'utiliser des filtres de bonne qualité. Au passage, on remarque que l'utilisation des mipmaps permet également de préserver la qualité des textures, tout en économisant la bande passante. Notez toutefois que les mipmaps prennent plus de place en mémoire, et qu'ils demandent du calcul supplémentaire à la puce pour déterminer le niveau de mip à employer.

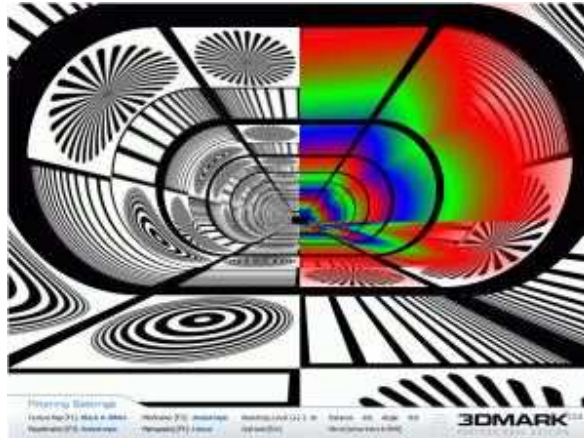
Seul problème, les dimensions des mipmaps sont divisées par deux à chaque fois (ex : texture de 256x256 pixels, puis 128x128 pour le mipmap suivant, puis 64x64, etc.). D'où la nécessité d'un filtrage, afin que les transitions deviennent invisibles y compris à longue distance.

Sans rentrer dans les détails, la différence de filtrage s'explique entre autres par le nombre de sample de texture utilisés au départ pour chaque pixel final – 4 pour le bilinéaire, 8 pour le trilineaire, 16 pour l'anisotropie 2x, etc. Le trilineaire consiste simplement à appliquer deux fois un filtrage bilinéaire, et à interpoler les deux points résultats.



Observons le résultat sur le tunnel de 3DMark03. L'écran se divise en 4 : en haut un filtrage bilinéaire est appliqué, en bas un filtrage trilineaire ; à gauche le rendu est normal, à droite les mipmaps ont été colorés afin d'y voir plus clair. Si l'on observe ces derniers, on se rend compte assez clairement que dans le cadre d'un filtrage bilinéaire, il n'y a pas vraiment de transition entre chaque texture de qualité différente, au contraire du filtrage trilineaire. Ainsi, en comparant maintenant le rendu à gauche, en bilinéaire on distingue facilement au premier plan les transitions entre mipmaps même si ceux-ci ne sont pas colorés. Ce n'est pas le cas du trilineaire.

Dès lors, quel est l'intérêt de l'anisotropie ? C'est très simple. En passant à chaque mipmap suivant, on perd en résolution, donc en détail et en précision : ceci génère un flou aisément observable. Ce flou est amplifié par l'effet de perspective, qui est utilisé par tous les jeux représentant une scène en 3 dimensions. L'anisotropie consiste donc à augmenter encore le nombre de texels sources, et à prendre en compte les déformations de l'objet sur lequel la texture est appliquée.



Ici, on retrouve le filtrage trilineaire en haut, et l'anisotropie 16X en bas. Au final, on arrive donc à supprimer complètement le flou.

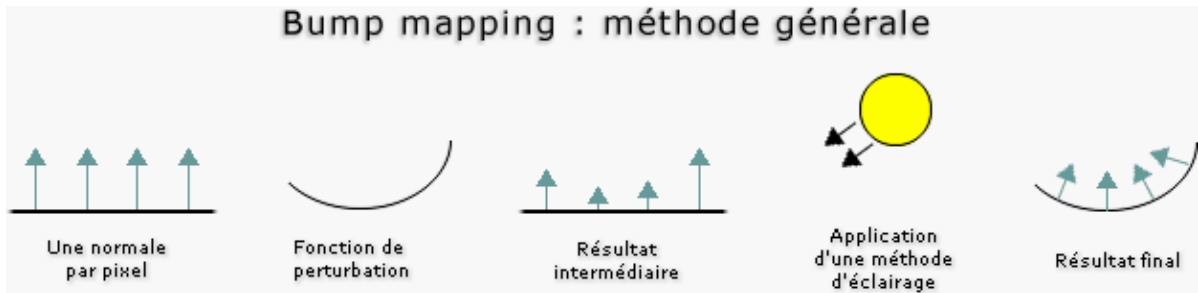
d) **Bump Mapping :**

Le bump mapping est une technique permettant de donner l'illusion de relief, sans augmenter la géométrie.

Quel rapport entre la géométrie et le bump mapping ? En théorie, dans un espace en 3 dimensions, seuls les objets en 3 dimensions peuvent avoir un relief. Or, une texture est un objet en 2 dimensions et les pixels qui la composent sont aussi en 2 dimensions. Pour obtenir du relief sur une texture, il faudrait transformer chaque pixel en élément de géométrie, c'est-à-dire en point ayant des coordonnées X,Y et Z.

Cette solution est inenvisageable dans la mesure où cela augmenterait d'un facteur 100 à 1000 le travail du processeur graphique et du microprocesseur.

Méthode générale du bump mapping



- 1 - On affecte à chaque point de la texture une normale.
- 2 - On applique aux normales de la texture une fonction de perturbation, qui correspond aux variations de la hauteur du relief qu'on veut donner à la texture.
- 3 - On applique une méthode d'éclairage où les normales de la surface subissent une modification de leur direction.

Premier exemple : Il s'agit d'une texture appliquée à 1 ou plusieurs polygones disposés dans le même plan. C'est le cas des murs, des plafonds ... Cette texture a subi une perturbation et l'application d'un modèle d'éclairage. On peut voir par contre que l'effet spéculaire est pris en compte.





Deuxième exemple : Il s'agit d'un objet complexe composé de plusieurs polygones et textures. L'ensemble a subi une perturbation pour donner l'impression de relief, un ombrage associé à un modèle d'éclairage. L'effet spéculaire n'est pas présent et pourrait laisser penser qu'on utilise de l'éclairage, mais il ne faut pas oublier que les reflets spéculaires ne sont pas présents sur toutes les surfaces, seules les surfaces moyennement ou fortement réfléchissantes peuvent avoir un tel effet, ce qui n'est pas vraiment le cas de la Lune.



e) S3TC ou DXTC :

La compression des textures dans un jeu a deux objectifs. Le premier est de réduire la place que peuvent occuper les textures. Le second est d'augmenter la vitesse de traitement des textures par la carte graphique.

1 - Réduire la place occupée par les textures :

Le fait qu'une carte graphique dispose de sa propre mémoire permet à celle-ci de traiter beaucoup plus rapidement les textures qui y seront stockées. D'une part, parce que ces mémoires sont souvent beaucoup plus rapides que celles de la mémoire centrale de l'ordinateur, et d'autre part, le processeur graphique peut y accéder directement sans passer par des intermédiaires.

Si la présence de mémoire assure donc un bon niveau de performances, elle implique aussi une limite quant à la quantité de textures que la carte graphique peut gérer de façon autonome. Si cette limite est dépassée, alors le processeur graphique doit faire appel à la mémoire RAM centrale de l'ordinateur avec pour conséquence une chute des performances.

L'objectif de la compression des textures est justement d'assurer que même avec une quantité de textures supérieure aux capacités de stockage de la carte graphique, celles-ci puissent être stockées une fois compressées.

2 - Vitesse de traitement des textures :

Le principal goulet d'étranglement dans les performances d'une carte

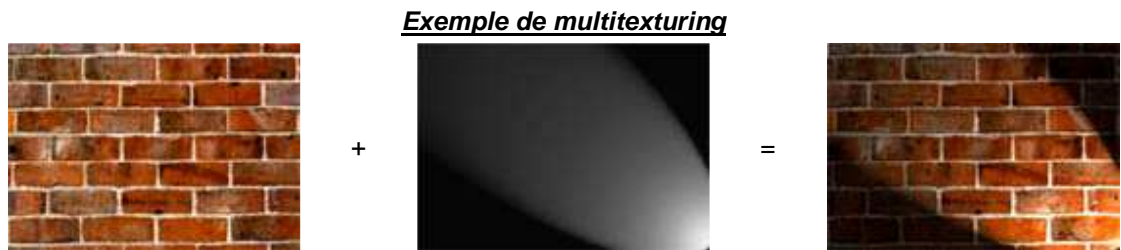
graphique se situe dans la quantité d'informations échangées en permanence dans le processeur et la mémoire de la carte graphique. En compressant les textures, on diminue la quantité de ces échanges tout en obtenant le même résultat. Autrement dit, dans un temps donné, une carte graphique va pouvoir gérer plus de textures si elles sont compressées que si elles ne le sont pas. La conséquence directe est une augmentation de la vitesse d'affichage des jeux. En moyenne, ce gain est de l'ordre de 5% à 10%.

La compression de textures est aujourd'hui devenue une chose courante, voire implicite dans la plus part des applications 3D, surtout dans les jeux. Même si elle n'est plus au centre des discussions sur les dernières cartes graphiques, il ne faut pas perdre de vue que sans elle les graphismes de jeux tels que Unreal Tournament 2003 ou IL2 Sturmovik n'auraient pas été possibles sur des systèmes avec 32 à 64 MB de mémoire vidéo. La compression de textures est aussi indispensable à l'essor des textures 3D qui prennent une place mémoire imposante et qui commencent tout doucement à apparaître sur nos PC.

f) Multi-texturing :

Elle est très utilisée dans les jeux. Cette technique applique plusieurs textures à un même objet au même moment, permettant l'ajout d'effets de lumière et de reflets. Il s'agit d'un procédé très populaire. La plupart des circuits graphiques modernes peuvent traiter simultanément deux textures sur un même objet, et non plus successivement comme auparavant.

Au lieu d'utiliser une seule texture, on peut décider d'en appliquer plusieurs pour créer des effets de lumière par exemple.



Dans cet exemple, 2 textures sont mélangées (blending) mais rien n'empêche d'en utiliser 3, 4, voir plus ... Bien évidemment, plus on augmente le nombre de textures, plus on utilise de Texels.

Les données passent soit par l'unité de multitexturing, soit par l'unité qui traite les pixels shaders. Le multitexturing, comme son nom l'indique, permet d'appliquer une ou plusieurs textures sur une surface. Les cartes 3D "modernes" permettent d'appliquer plusieurs textures d'un seul coup (on dit "en une seule passe") au même objet. Par exemple, une GeForce3 permet d'habiller un objet avec quatre textures en une seule passe. C'est grâce au multitexturing qu'on peut utiliser la technique du lightmapping, qui permet d'obtenir des résultats très réalistes. Cette méthode consiste à utiliser un jeu de textures pour donner son aspect général à un objet, et un autre jeu pour lui appliquer des effets de lumière précalculés.

Si les vertex shaders permettent une plus grande flexibilité dans la manipulation de la géométrie des objets, les pixels shaders sont leurs équivalents pour les pixels. On peut grâce à eux changer la façon dont les pixels sont sélectionnés, mélangés et rendus. Cela permet par exemple de créer des effets comme la gestion de l'éclairage par pixel (plutôt que de la gérer par vertices, puis d'interpoler) ou le bump-mapping.

Si un brouillard a été demandé par le programme, c'est une fois le multitexturing ou le pixel shading achevé qu'il sera ajouté. Utile pour créer des ambiances, (montages embrumés, rue sombres...) il est

cependant souvent utilisé pour limiter la profondeur de champ d'une scène, de façon à économiser des ressources en n'affichant pas les objets lointains, masqués par le brouillard.

g) Le DirectX de Microsoft et les versions de modèle de shader :

Ces termes sont probablement les plus souvent cités et les moins souvent compris. Pour saisir leur signification, une courte explication et un bref historique des API **graphiques** est indispensable. DirectX et OpenGL sont des API graphiques. API est l'acronyme de « Application Programming Interface », ce qui signifie essentiellement une norme de programmation accessible à tous.

Avant les API graphiques **3D**, tous les constructeurs de cartes graphiques avaient leur propre méthode propriétaire pour faire fonctionner leurs cartes avec les jeux. Des développeurs étaient contraints de programmer les jeux avec les lignes de code spécifiques à chaque constructeur pour chaque type de cartes graphiques qu'ils voulaient supporter. C'était une méthode très coûteuse et inefficace. Pour résoudre ce problème, les API graphiques 3D ont été créées, de façon à ce que les développeurs puissent programmer leurs logiciels pour être compatible avec l'API et non plus avec chaque matériel spécifique. La responsabilité de la compatibilité est alors revenue aux constructeurs de cartes graphiques qui devaient faire en sorte que leurs drivers soient compatibles avec l'API.

La seule complication est que deux API différents ont émergé, tous les deux étant utilisés aujourd'hui. Les deux Api qui comptent sont DirectX de Microsoft et la norme OpenGL, GL étant les initiales de « Graphics Library » (bibliothèque graphique). Comme l'API DirectX a une influence beaucoup plus forte sur la technologie matérielle du **processeur graphique** en ce qui concerne les jeux, nous allons nous concentrer dessus. Il est par ailleurs actuellement plus important pour les jeux.

DirectX est une création de Microsoft. En réalité, DirectX est un ensemble d'API dont une seule est pour le graphisme 3D. DirectX inclut des API pour le son, la musique, les périphériques d'entrée et les médias, pour en citer quelques-uns. L'API DirectX spécifique qui s'applique au graphisme 3D s'appelle Direct3D, mais quand on parle de cartes graphiques on admet généralement que les termes DirectX et Direct3D soient employés de façon interchangeable.

DirectX est périodiquement mis à jour dès lors que la technologie matérielle du graphisme progresse et que les développeurs de jeux améliorent les méthodes qu'ils utilisent pour programmer les jeux. Comme DirectX a rapidement gagné en popularité et en utilisation, les constructeurs de **processeurs graphiques** ont commencé à concevoir leurs produits pour qu'ils collent aux plus récentes capacités de DirectX. Pour cette raison, les cartes graphiques sont souvent décrites par leur version de DirectX (versions DirectX 8, 9.0, ou la toute dernière 9.0).

Pour compliquer un peu plus les choses, des éléments de l'API Direct3D peuvent changer selon des paliers plus petits que ceux de l'ensemble DirectX en entier. La spécification DirectX 9.0, par exemple contient le support pour Pixel Shader 2.0. Cependant, DirectX 9.0c contient la spécification Pixel Shader 3.0. Cela signifie que toutes les cartes graphiques DirectX 9 ne supportent pas les mêmes fonctions ; la Radeon 9700 supporte supports Shader Model 2.0, et la Radeon X1800 Shader Model 3.0, même si toutes les deux peuvent être décrites avec précision comme des « cartes graphiques DirectX 9 ».

Il faut garder à l'esprit que les développeurs pensent toujours aux possesseurs de matériel ancien quand ils créent un **jeu**, car les ignorer limiterait sévèrement leur marché. Dès lors, les développeurs programment généralement des « lignes de code » séparées dans un jeu pour supporter le matériel ancien. Un jeu DirectX 9 peut avoir des lignes de code DirectX 8 et même DirectX 7 pour permettre la compatibilité. En règle générale, ces anciennes lignes de code font

l'impasse sur les fonctions visuelles que peuvent afficher les cartes graphiques modernes, mais elles permettent du moins aux vieilles cartes graphiques de faire tourner le jeu.

Beaucoup de nouveaux jeux ont besoin que la dernière version de DirectX soit installée pour tourner, même si la **carte graphique** sur laquelle tournera le jeu est ancienne. Un jeu tout nouveau qui n'utilise que la technologie DirectX 8 peut avoir besoin de la dernière version de DirectX 9 pour tourner, même sur du matériel de classe DirectX 8.

Maintenant que vous possédez les connaissances de base sur ce qu'est DirectX 8, quelle sont les différences entre les différentes versions de l'API Direct3D API dans DirectX ? Les premières versions de DirectX (3,5, 6 et 7) étaient relativement simples en ce qui concerne l'API Direct3D. Les développeurs devaient mélanger et faire se correspondre des effets visuels d'une liste immuable d'effets pré programmés. Le grand bond vers la programmation graphique moderne s'est effectué avec DirectX 8. DirectX 8 a lancé la programmation de shader personnalisée, ce qui a permis aux développeurs de créer pour la première fois des effets visuels véritablement personnels. DirectX 8 supportait de Pixel Shaders 1.0 à 1.3 et Vertex Shader 1.0. DirectX 8.1 est une version améliorée de DirectX 8 qui supporte Pixel Shader 1.4 et Vertex Shader 1.1.



Dans DirectX 9, la complexité du shader a été étendue. DirectX 9 supporte Pixel Shader 2.0 et Vertex Shader 2.0. DirectX 9c est une version améliorée de DirectX 9 qui inclut la spécification Pixel Shader 3.0.

DirectX 10 est la future version qui accompagnera la prochaine version de Windows, Windows Vista. Il ne sera pas compatible avec Windows XP.

X. OPERATIONS POSSIBLES DANS UN PIXEL SHADER

Le Pixel Shader permet quant à lui de calculer la couleur de tout point de la surface de rendu couvert par une des primitives que nous avons envoyées à notre carte graphique. Par exemple dans le cas où nous envoyons à la carte une liste de triangles, nous allons en fait lui envoyer une liste de sommets et une liste d'indices précisant comment ces sommets doivent être reliés entre eux pour constituer les triangles désirés. Tous les sommets vont être transformés par notre Vertex Shader ce qui abouti à la même liste de points mais cette fois positionné dans le cube $(-1, -1, 0)$ $(1, 1, 1)$ résultant de la transformation du Viewing Frustrum par la matrice de projection. Les points en même tant que la liste d'indice sont ensuite passés dans le pipeline à ce qui est appelé dans notre diagramme « Primitive Processing ». Pour résumer, les traitements regroupés sous ce nom récupèrent la liste d'indices afin d'en déduire les sommets qui forment triangles et autres primitives que nous aurions pu passer à la carte.

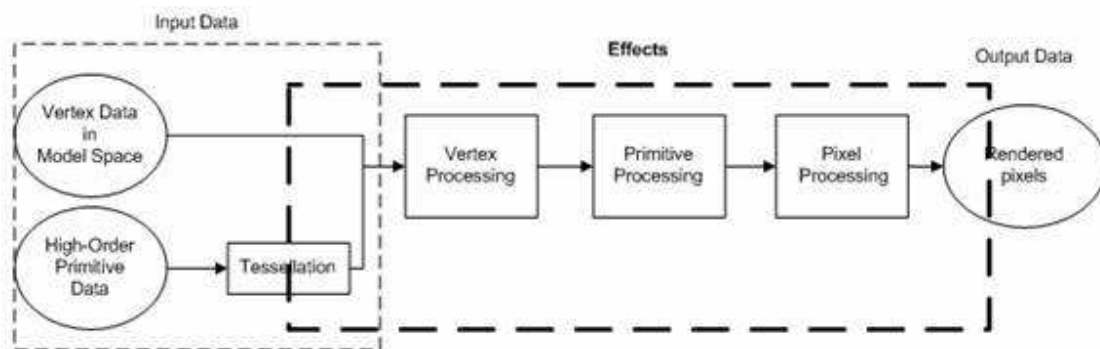
Ensuite cette unité de traitement génère et passe au Pixel Shader les points de la surface de rendu couvert par cette primitive en fonction des informations qui lui sont passés par le biais de la structure VS_OUTPUT.

Nous voyons donc bien que le Pixel Shader nous permet de jouer directement, de calculer la couleur des points affichés à l'écran (ou tout du moins certains d'entre eux). Là encore un grand champ d'opérations différentes peut être effectué. On peut à nouveau se reporter à la section intitulée « HLSL Intrinsic Functions » de la documentation du SDK pour bien cerner ce qui est possible. Dans la suite de cet article je me propose de modifier l'exemple de mon premier document (DirectX Managed : Rappel des concepts de bases) pour obtenir une eau et un ciel un tout petit peu plus crédible que ce qu'ils étaient initialement.

Par exemple, pour le ciel, plusieurs textures seront utilisées et mélangées par le Shader. Les coordonnées de textures seront calculées à la volée pour animer le positionnement des textures plaquées sur notre ciel. Pour l'eau aussi, plusieurs textures seront utilisées, ainsi que le contenu du back Buffer pour obtenir une eau qui soit à la fois réfléchive et réfractives.

XI. Le 'Pipeline' graphique

Le terme anglo-saxon pipeline donne une idée de suite de traitements exécutés les uns après les autres. Le fonctionnement d'une carte graphique et les opérations qu'elle effectue pour obtenir le résultat visible à l'écran peut de fait être découpé en étapes successives. Si l'on se rapporte au schéma ci-dessous extrait de la documentation du SDK DirectX il donne une vision de très haut niveau de cette suite d'opérations.



Pour obtenir l'affichage d'objet en 3 dimensions à l'écran, il est nécessaire de fournir à la carte graphique par le biais d'un API graphique comme DirectX (ou OpenGL) une suite de primitives graphiques. Ces primitives sont elles-mêmes une suite de ligne ou de triangles, des textures peuvent être plaquées sur ces primitives et vont impacter la couleur finale de celles-ci à l'écran. Si on fait le bilan on fournit donc à notre carte des points de l'espace, des tableau d'indices permettant de relier ces points et constituant les primitives, des textures. Les opérations effectuées sur ces données apparaissent dans le schéma ci-dessus.

Si l'on regarde les différentes étapes plus en détails, d'abord les points (les vertex contenus dans le VertexBuffer et passés à la carte) sont transformés en fonction des matrices 4x4 qui leur sont associées ainsi que des matrices View et Projection décrites dans mon document précédent. Dans le schéma ci-dessus il s'agit de l'étape appelée « Vertex Processing ».

Ensuite les primitives sont reconstituées à partir de tableau d'indices qui sont eux aussi passés à la carte et qui définissent les indices des points dans le tableau de points précédemment traités qui constituent par exemple les sommets de nos triangles. Une fois les primitives à affiché reconstituées, on détermine les portions de la surface de rendu couverts par telle ou telle primitive graphique, la couleur aux sommets ou des extrémités des primitives est calculé et extrapolée pour tout les autres points de la primitive tout en tenant compte de la texture que l'on a décidé d'appliquer, il s'agit de l'étape appelée « Primitive Processing » (cf. diagramme ci-dessus).

On voit bien que la suite d'opérations décrite ici est relativement fixe. En particulier, même si je ne donne pas plus de détails sur la façon dont la couleur en tout point d'une primitive est calculé en fonction de la couleur aux sommets et de la texture, dans le 'pipeline' décrit ici, cette opération est systématiquement la même quelques soit la primitive en cours de traitement. Grosso modo il s'agit peut ou prou d'un mode de fonctionnement commun à toute carte graphique dite 3D et à la suite d'opérations effectuées par la carte pour afficher les objets. Ce mode de fonctionnement, aussi abouti soit-il pour afficher un certain type d'objets 3D, peut se révéler contraignant dans la mesure où il est inaltérable (et ce même si certaines variables permettent toutefois de jouer sur la façon dont ces opérations pré déterminées vont être exécutées et ainsi de modifier le résultat final). Dans toute la littérature anglo-saxonne traitant du sujet il est fait référence à ce pipeline sous l'appellation de Fixed-Function Pipeline.

C'est pour lever cette limitation que les Shaders ont fait leur apparition. Au lieu d'appliquer à tout point de l'espace en 3D puis à tout pixel de la surface de rendu couvert par une primitive une opération pré déterminée et gravée dans le silicium des processeurs graphiques, les cartes modernes offrent la possibilité d'exécuter une suite d'opérations spécifiées par le programmeur par le biais de programmes qui seront exécutés directement par le processeur graphique, nos fameux Shaders.

a) **Rendering :**

Il s'agit de l'ultime étape du calcul d'une image avant son affichage. Le rendering finalise les pixels qui seront effectivement visibles à l'écran, en fonction des textures employées et des effets de lumière, d'ombre ou de brouillard.

La liste des fonctions spécialisées 3D est encore très longue. Au fur et à mesure des années et des technologies, le nombre de ces fonctions augmentera sans cesse. Le nombre de combinaison pour générer le plus parfaitement possible un objet quelconque est infini. Mais une chose est sûr, le chipset ou l'accélérateur graphique devra supporter ces fonctions et devra sûrement les intégrer. Mais avant tout, il devra être capable de calculer toutes ces fonctions dans des temps acceptables.

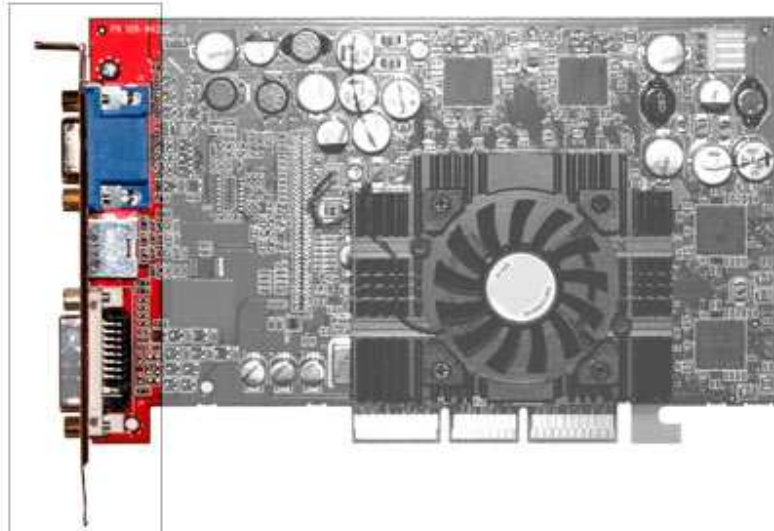
XII. BIOS Vidéo

Au même titre qu'un PC, une carte graphique possède son propre BIOS. Celui-ci est spécifique au fabricant de la carte. Ceci explique des différences de performances entre 2 cartes de Chipset identiques. C'est lui qui contient tous les paramètres de la carte graphique, certains pourront être réglés par l'utilisateur, affichage modes couleurs, résolution, fréquence d'affichage de l'image à l'écran, réglage de l'image à l'écran (luminosité, contraste, Gama), réglage des canaux de couleurs actifs(c'est à dire régler indépendamment les tons des couleurs RVB rouge vert bleu)

Tout comme le BIOS de la carte mère certains BIOS de carte graphique pourront être Flashable et donc pouvoir être rajeunis et recevoir de nouveaux des paramètres plus sophistiqués et à l'ordre du jour.

XIII. Connecteur de sortie

A. Les sorties



C'est là que se trouvent les sorties de la **carte graphique**. A noter qu'un côté de presque toutes les cartes d'extension est accessible par l'arrière du **PC** et l'on trouve donc une plaque métallique qui supporte les différents connecteurs.

Quand une carte graphique est installée dans un PC, on ne voit donc les connecteurs qu'à l'arrière de celui-ci. C'est dans cet élément que l'on branche le câble de l'écran. Beaucoup de **cartes graphiques** ont plusieurs sorties (deux), de telle sorte que l'on peut utiliser plus d'un écran en même temps. Il existe de nombreuses interfaces d'affichage, mais les interfaces d'affichage principales d'un PC sont soit numérique, soit analogique.

Les PC sont des machines binaires qui traitent des zéros et des uns en tant qu'éléments binaires. Par conséquent, le numérique est la sortie native de la carte graphique. Les écrans modernes viennent d'une longue lignée d'écrans à tube cathodique (CRT). Un écran CRT utilise un **canon** à électron pour tirer trois matériaux différents à l'intérieur d'un tube qui émet des lumières rouges, vertes et bleues quand on l'excite.

Ces premiers appareils étaient analogiques par nature et pour convertir de numérique à analogique un élément appelé convertisseur numérique vers analogique (DAC, acronyme anglais de « Digital Analogic Converter ») a fait son apparition dans les sorties graphiques. Avec l'avènement des écrans numériques comme les écrans à cristaux liquides (LCD), il y a de moins en moins besoin d'un DAC, mais ce composant est toujours incorporé pour le support analogique.

a) Les sorties VGA (D-Sub)



Le prise écran analogique a 15 connecteurs et peut être identifiée par sa couleur bleue.

Si l'on parle du VGA en tant que résolution, c'est l'acronyme de « Video Graphics Array » (ce qui consiste en un certain nombre de pixels horizontaux et verticaux). Dans le domaine du matériel graphique cependant, c'est l'acronyme de « Video Graphics Adapter ». Le connecteur correspondant s'appelle D-Sub 15 et il achemine un signal d'affichage analogique, dont la qualité peut varier en fonction de chaque produit. Les cartes graphiques les plus chères doivent être capables de fournir un signal propre qui donne un affichage bien piqué à hautes résolutions.

Cette **interface** était la sortie standard avant que le DVI numérique (Digital Visual Interface) ne fasse son apparition et elle est toujours très courante. Les sorties VGA D-Sub peuvent se brancher à la plupart des écrans CRT (à tube) classiques. Elles peuvent aussi se brancher à la plupart des projecteurs numériques et même à certaines HDTV (téléviseurs **haute définition**), même si nous déconseillons de les utiliser avec n'importe quel type d'écran numérique pour une question de qualité d'image.

b) Les sorties DVI



DVI est l'acronyme de « Digital Video/Visual Interface »

le DVI est la sortie numérique standard des cartes graphiques et des écrans plats (à l'exception des modèles à bas prix). Si vous possédez un modèle pas plus ancien que 2004, il y a des chances qu'il ait une sortie DVI. La plupart des cartes graphiques avec sorties DVI sont accompagnées d'un adaptateur pour convertir le DVI en VGA/D-Sub pour ceux n'ayant pas d'écran numérique. Toutes les cartes graphiques haut de gamme ont deux ports DVI pour connecter deux écrans et étendre le bureau Windows sur les deux. Cela dit, avec deux sorties, que ce soit une D-Sub et une DVI, ou deux sorties du même type, le mode d'affichage à deux écrans est supporté. Les nouveaux écrans numériques comme les écrans 30 pouces de **Dell** et Apple nécessitent une carte graphique avec deux sorties graphiques et deux câbles pour afficher à leur résolution native de 2560 x 1600.

c) La vidéo composite



Voici la prise jack Vidéo composite aussi appelée RCA (Radio Corporation of America).

C'est la sortie vidéo traditionnelle utilisée communément pour les **télévisions** et les appareils s'y rapportant, comme les VCR (**magnétoscopes**). Le signal vidéo voyage à travers un seul jack de style coaxial. Cela fournit un signal analogique de très faible résolution utilisé classiquement pour les présentations et les jeux. Il ne convient

pas vraiment pour lire n'importe quoi à cause de la pauvreté du signal et de la qualité d'image. Il convient cependant pour afficher les vidéos de définition standard.

d) S-Video (ou S-VHS)



S-Vidéo désigne la « Super Video » ou le « Super VHS ».

Le S-vidéo est une norme de vidéo analogique utilisée par l'industrie de la **télévision**. Il fournit un signal basse résolution aux télévisions comme la vidéo composite à un seul **câble**, car toutes les informations de couleurs sont dans un seul signal, mais il sépare la luminosité et les couleurs. Il y a quatre aiguilles et quatre fils dans un câble S-Vidéo standard dont un pour la luminosité, et un autre pour sa terre, et un pour les couleurs et un autre pour la terre correspondante. Il offre un signal de meilleure qualité que la vidéo composite à un seul câble, mais toujours à une résolution à dynamique basse. Cela dit, bien que le S-VHS soit supérieur à la vidéo composite à un seul câble, il est largement inférieur aux sorties vidéo de haute qualité (Y, Pb, Pr).

e) La vidéo composite



Les prises de la vidéo composite sont trop grosses pour être sur la carte elle-même et se trouvent généralement sur un dongle. Ce dongle possède à la fois des composants vidéo (les trois premiers connecteurs) et audio (les deux derniers connecteurs).

Cette sortie utilise trois câbles coaxiaux de style jacks appelés "Y", "Pb" et "Pr". Ils fournissent des informations de couleurs séparées pour la HDTV (**télévision haute définition**). Ce type de sortie se raccorde aussi à la plupart des projecteurs numériques. Même si le signal est analogique, c'est un signal de qualité, séparé, et de haute résolution comme en fournissent les sorties VGA. Les contenus HD ou « Haute Définition » peuvent être acheminés via la vidéo composite.

XIV. **HDMI**

HDMI est l'acronyme de "High Definition **Multimedia Interface**"

Le HDMI est la future norme. C'est la seule sortie qui supporte à la fois l'audio et la vidéo dans un même câble. Le HDMI a été créé pour les applications télévision/cinéma, mais comme c'est un format très sûr, les utilisateurs d'ordinateur en auront besoin pour voir les futurs contenus **haute définition** dans un format haute résolution.

Les prises HDMI sont encore très rares sur les cartes vidéo, mais on s'attend à ce qu'elles deviennent bien plus courantes à l'avenir. Regarder de la vidéo haute définition sur un PC peut nécessiter à la fois une **carte vidéo** HDMI et un écran compatible HDMI.

xv. Conclusion

Pour conclure, nous pouvons dire que la carte graphique est un composant clé pour que l'ordinateur puisse fonctionner correctement. Nous avons pu voir que la carte vidéo est composée de plusieurs éléments ayant chacun des fonctions très spécifiques et primordiales. Parmi ces composants nous retrouvons le chipset, le Ramdac, la mémoire vidéo, le type de bus utilisé et bien sûr les connecteurs d'entrées/sorties.

Cependant nous avons pu constater que la carte graphique ne se limitait pas seulement au traitement des informations à envoyer sur un moniteur. Ceci était vrai il y a encore quelques années. De nos jours, les cartes graphiques sont devenues de véritables centres de traitements et de calculs pour soulager le processeur principal de l'ordinateur. On parle même d'accélérateur graphique. Des centaines de fonctions préprogrammées sont implantées dans les chipsets. Ainsi, les cartes vidéos ont désormais le rôle d'effectuer tous les calculs nécessaires à la modélisation d'un objet 3D. Lors de l'affichage, cet objet 3D doit avoir un aspect proche de la réalité. Aussi, nous avons vu que pour exploiter toute la puissance de ces cartes graphiques, il était nécessaire de disposer des pilotes adéquates. En plus de ces pilotes, il existe d'autres programmes dit A.P.I permettant d'utiliser toutes les ressources de la carte dans des logiciels de graphismes ou de jeux.

Actuellement, les cartes vidéos évoluent à un rythme très soutenu. Les performances et les fonctions supportées sont de plus en plus accrues. Si nous tenons à rester à la "mode", il faudrait changer sa carte vidéo tous les 6 mois afin de bénéficier des dernières avancées technologiques. A quand la carte vidéo universelle qui intégrera et calculera toutes les fonctions requises pour un affichage 3D hyperréaliste ?

