

[PC](#)
[Nintendo](#)
[Playstation](#)
[XBox](#)
[Oculus](#)
[Mobile](#)
[Retro](#)

f (<http://www.facebook.com/standalonepost>)



[_ \(http://feeds.feedburner.com/jeuxDotVideo\)](http://feeds.feedburner.com/jeuxDotVideo)



ACCUEIL

EN BREF

EMISSIONS

ARTICLES

TESTS

DOSSIERS

CONSOLES

PORTAIL JV ([HTTP://PORTAIL.JEUX.VIDEO/](http://portail.jeux.video/))

Les textures, une histoire de style

14 Avril 2014 Thomas Daveluy 5004 lectures



Quels sont les points communs entre *Far Cry 3* et *Super Mario 64* ? Tous deux sont constitués des mêmes éléments : des polygones et des textures. Si l'arrivée de la 3D a bousculé la manière de faire des jeux, et que les moteurs graphiques ne cessent d'être de plus en plus performants, la technique reste la même depuis ces 20 dernières années. L'évolution technologique des consoles a permis d'augmenter de manière exponentielle le nombre de polygones affichés à l'écran, mais c'est très certainement la possibilité de fournir des textures de plus en plus travaillées, avec de plus en plus d'effets, qui a permis au jeu vidéo une telle progression dans son évolution graphique. Mais les textures ne sont pas juste une manière de faire illusion. Il s'agit aussi, dans de très nombreux jeux, de l'élément qui définit tout l'univers graphique et donc le style de l'oeuvre.

Si *Wolfenstein 3D*, un des premier jeu tridimensionnel, permettait effectivement de se déplacer dans la profondeur, **on ne peut pas dire que cet opus soit réellement un jeu tridimensionnel**. En vérité il utilisait une technique d'images planes tournées vers la caméra dont la taille dépend de l'éloignement : **ce que l'on appelle des sprites**. Il est impossible de faire le tour d'un ennemi, il est constamment face à la caméra. Il faudra attendre des jeux comme *Starwing*, sorti sur Super Nintendo en 1993, pour voir des objets réellement en 3D, c'est à dire composés de faces qui forment un ensemble autour duquel le joueur peut tourner.



Wolfenstein 3D

Starwing sur Super Nintendo

La rasterisation, un peu de technique...

Le procédé utilisé en 3D temps réel est globalement le même sur les premiers jeux que sur ceux d'aujourd'hui. Un objet est composé de points (vertex) qui possèdent des coordonnées sur 3 axes (x, y, z). La liaison entre trois points^[1] forme ce que l'on appelle une face. Un objet 3D, nommé polygone, est alors composé d'un nombre plus ou moins important de faces. La machine doit procéder pour chaque image affichée (frame) à un calcul des coordonnées de chaque point pour dessiner les faces, **c'est ce que l'on appelle la rasterisation**.





Le procédé de rasterisation inclue aussi les calculs d'ombrages : une face est plus ou moins sombre selon son orientation. Il est possible de demander à la machine d'**appliquer une image sur une face**, un peu comme si on la collait dessus, le processus de rasterisation se chargeant de calculer les déformations de la texture en fonction de l'orientation.

Aujourd'hui le procédé paraît relativement habituel et **les jeux actuels peuvent gérer des millions de polygones et des centaines de textures** avec une palette d'effets très importante, mais cela n'a pas toujours été aussi simple.

Les premiers jeux 3D : quand le papier peint se décolle sur les bords

Du fait de la faible puissance de la SNES, *Starwing* ne permettait d'afficher qu'un nombre assez limité de faces et surtout **la console était incapable de calculer la moindre coordonnée de texture**. Les polygones de ce jeu sont donc uniquement colorés et vaguement ombragés. A l'époque il s'agit malgré tout d'une prouesse technique inégalée.

En 1994 arrive la PlayStation, **une console 32 bits suffisamment puissante pour appliquer des textures aux polygones**. La méthode est cependant un peu hasardeuse, les textures flottent, se déforment mal, voire disparaissent parfois.



Les textures avaient tendance à se déformer au premier et à l'arrière plan

Les déformations hasardeuses des textures étaient souvent dues à un problème de calcul. En gros **les moteurs de l'époque effectuaient des approximations** pour appliquer les déformations de la perspective. Ces calculs trop simplistes entraînaient des imprécisions très visibles sur la position des textures^[3].

Très présent sur la PS1 **ces bugs de texture disparaîtront vite avec l'apparition d'algorithmes de correction de perspective** et de nouvelles machines plus puissantes pouvant effectuer des calculs plus complexes. De nouvelles méthodes de filtrage permettront par ailleurs d'affiner le rendu des textures (filtrage bicubique, trilineaire ou anisotrope).

Dès lors le vrai travail de texture peut commencer.

Un peu de barbouillage

Si au départ **les textures étaient souvent issues d'objets photographiés** et vaguement retouchés, les développeurs de jeux se sont rendus compte qu'il était à la fois plus simple mais aussi beaucoup plus beau de présenter les détails sur les textures et non sur les objets. Ainsi **nul besoin de modéliser une éraflure** sur une arme, il suffit de la peindre et d'y appliquer un certain nombre d'effets.

Les rayures et saletés sur l'arme sont évidemment présentes sur le fichier de texture mais pas sur le modèle polygonal

Pour créer les textures d'un objet, les graphistes utilisent des outils (généralement disponibles directement dans les logiciels de modélisation 3D) pour **éclater le volume et obtenir une représentation à plat de cet objet**.



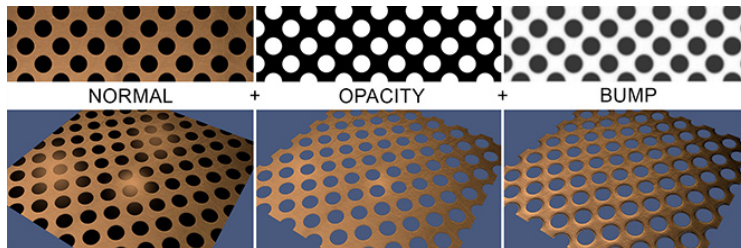
À partir de ce gabarit, les graphistes vont créer plusieurs fichiers de textures qui vont se compléter sur des calques, dont voici quelques exemples :



Un éclaté

- Un calque d'**image ambiante** (normal map) : il s'agit d'une image en couleur qui contient la texture sans aucun effet.
- Un calque de **relief** (bump map) : une image en noir et blanc qui va forcer les reflets et la lumière à se déformer, comme s'il y avait des bosses ou des creux sur l'image.
- Un calque d'**opacité** (transparency map) : une image en noir et blanc qui va définir la transparence.
- Un calque de **reflet** (reflect map) : une image en noir et blanc

On parle de **reflet** (reflect map) : une image en noir et blanc qui définira les zones qui agiront comme des miroirs. (Cette technique n'est pas utilisée dans les jeux en temps réel car trop gourmande en ressources^[2])



Il s'agit évidemment d'une liste non exhaustive. Les fichiers de texture ainsi que le nombre de superpositions peuvent varier en fonction du moteur graphique utilisé.

Le plus gros de ce travail va donc consister à **peindre des ombres ou à souligner des reflets de manière totalement artificielle**. C'est une technique que l'on retrouve au début du cinéma, notamment dans les films expressionnistes allemands, où l'ensemble des décors sont créés à partir de planches sur lequel sont peints des décors et des ombres.

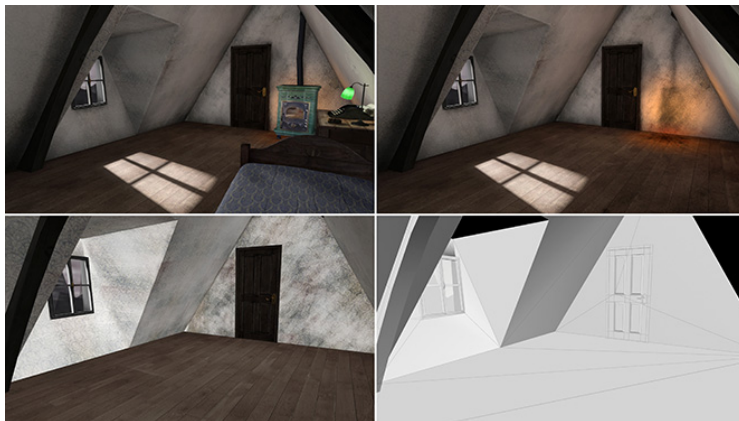


Le cabinet du Dr. Caligari, 1920

Prenons pour exemple les 4 images ci-dessous. Le passage entre la deuxième vignette et la 3ème se fait en **enlevant les calques d'ombres et de lumières directement depuis le fichier de textures**. Aucune lumière ni effet n'est présent.

L'élaboration d'un jeu en 3D passe donc par l'emploi de graphistes 2D qui seront chargés de travailler les textures du jeu en imitant les effets de lumière, de déformation et de relief. Sur certaines grosses licences, **les graphistes 2D ne côtoient même pas ceux qui travaillent à la modélisation des objets**. Ils reçoivent un cahier des charges, un gabarit et fournissent un ensemble de fichiers répondant à la demande.

Tous ces effets ont été créés dans l'unique volonté **d'imiter au plus près les phénomènes physiques que l'on retrouve dans la nature**. Au fil du temps les moteurs graphiques se sont donc dotés d'outils et de palettes d'effets permettant de simuler au maximum un environnement réaliste. Cependant les graphistes se sont aussi rendu compte que **ces outils pouvaient tout aussi bien tirer partie de leur**



puissance pour créer des images surprenantes, avec des styles diamétralement opposés à la recherche du réalisme

Ainsi naissent les styles

Comme pour la peinture ou le dessin, le jeu vidéo a su très rapidement tirer parti de la possibilité de quitter le réalisme pour présenter de nouvelles formes graphiques et ainsi créer de nouveaux genres, souvent empruntés à d'autres courants artistiques.



Jet Set Radio, un jeu en Cel-Shading

Lorsque la Dreamcast sort en 1999 (1998 au Japon) elle s'accompagne de quelques jeux inédits dont *Jet Set Radio*, un des tous premiers jeux à présenter un rendu **cartoon** : les contours sont soulignés par des traits et les ombrages sont plats, comme dans un dessin animé ou dans la BD. **De nombreux jeux par la suite emprunteront ce procédé nommé Cel-Shading** (ou Toon Shading ou encore, dans sa version franchouillarde, Ombrage de Celluloid) : *Zelda Wind Waker*, *Killer 7*, *Dragon Ball Z Budokai*, etc.

Une version plus poussée du Cel-Shading est apparue sur certains jeux, comme *Okami* qui s'inspire des fresques japonaises ou *Zelda Skyward Sword* dont **l'esthétisme est directement emprunté au mouvement impressionniste** en peinture.





L'univers de Zelda Skyward Sword, avec son rendu sous forme de taches, ressemble à une peinture impressionniste.

Il y a aussi les jeux qui proposent **des expérimentations graphiques en lien avec leur gameplay**. Le parti pris est directement lié à la manière dont le joueur va pouvoir se déplacer. C'est le cas des jeux comme *Minecraft* avec son monde en voxel, *Fez*, un jeu de plateformes mi-2D mi-3D, *Evoland* où les graphismes changeront au fil de l'aventure, *Journey* avec son univers contemplatif, ou encore *The Unfinished Swan* où **le joueur doit avancer dans un décor blanc en révélant l'architecture avec de la peinture noire**. La liste est beaucoup trop longue



pour tenter d'en faire le tour mais de plus en plus de jeux ces dernières années choisissent un style graphique réellement original.

Et demain ?

Avec les moteurs de rendus présentés ces dernières années il est de plus en plus clair que **la course au photoréalisme est pratiquement terminée**. À l'instar de l'hyperréalisme en peinture, un courant artistique qui a eu lieu dans les années 50 et qui est **vite tombé en désuétude**, une fois l'imitation parfaite atteinte, on ne peut plus vraiment s'attendre à de véritable innovation. Et c'est là que les choses deviennent intéressantes : il est pratiquement certain que **les prochaines années seront passionnantes pour le jeu vidéo** tant la recherche de nouveaux modes de représentation et la puissance des machines ouvriront d'avantage le champ des possibles.

[1] Beaucoup de vues 3D présentent des polygones à 4 points. En vérité il s'agit d'un raccourci avec la fusion de deux triangles, même si en vue filaire la liaison entre les points opposés n'est pas forcément représentée.

[2] Les reflets sont obtenus par un rendu en raytracing (lancer de rayon). Le moteur de rendu doit calculer pour chaque pixel son orientation pour réfléchir la zone opposée à la caméra. Quelques moteurs expérimentaux proposent du raytracing en temps réel, mais les machines n'ont pas encore la puissance nécessaire. Pour l'instant les développeurs trichent avec une projection map : ils utilisent soit une caméra invisible qui projette ce qu'elle filme soit un motif lui aussi projeté sur l'objet.

[3] Merci à [+Manuel Masiello](#) pour les corrections sur cette partie.

Tags:

[graphismes](#) [3d](#) [images](#) [textures](#) [cel-shading](#) [réalisme](#)
[polygones](#) [style](#) [graphique](#)



Auteur : [Thomas Daveluy](#)

Spécialiste Game Design

Après une formation scientifique, Thomas a effectué un virage à 180° pour s'engouffrer dans un cursus artistique. Cinq années d'école d'art l'ont alors convaincu que le jeu vidéo était une forme qui méritait ses lettres de noblesse mais qui était encore mal accepté par l'Académie. Sans cesse partagé entre l'art et la science, Thomas a trouvé dans le jeu vidéo le terrain d'expérimentation idéal : quel meilleur média que celui combinant des bases très techniques et un champ artistique à part entière ?

Derniers articles

- [Zombie Hunter Inc. en Early Access](#)
- [Chaperon Rouge - Les amants maudits](#)
- [Pourquoi le nouveau DOOM va t-il \(encore\) me décevoir ?](#)