

La programmation

Article scanné dans le magazine Gameplay 64 n°HS1 (Année 1997)

Sujet de l'article : Nintendo 64 (*)

Scans réalisés par les membres du site Nintendo64EVER,
usage exclusivement destiné aux autres membres du site.

Toute reproduction, partielle ou complète, ainsi que la diffusion de ce fichier est interdite.

Les magazines originaux sont la propriété intellectuelle exclusive de leurs éditeurs respectifs,
les scans regroupés dans ce fichier ont un but uniquement documentatif et informatif,
aucune exploitation commerciale ne peut en être faite.

La programmation

d'un jeu

Nintendo 64

Dossier réalisé par Stéphane

Ne vous êtes-vous jamais demandé ce qui se cachait derrière un jeu Nintendo ? Que peut-il bien y avoir dans une si petite cartouche pour nous proposer des jeux aussi grandioses ?

Comment la N64 est-elle capable de tant de prouesses ?

Bref, de quelle manière programme-t-on un jeu sur Nintendo 64 ? Toutes les réponses à vos questions sont dans ce dossier, qui n'aurait pas vu le jour sans l'aide d'Emmanuel Carré, Julien Merceron et Olivier Rozier, trois stars de chez Ubi Soft.

Conception

Au cOMmencEMeNt...

Que ce soit sur Sega, Saturn ou Nintendo 64, le début de la conception d'un jeu est le même : trouver une idée première, quelque chose qui puisse intéresser à la fois les éditeurs et les joueurs. Mais bien sûr, l'idée de base ne suffit pas : elle demande à être développée. Il faut trouver une histoire avec un ou plusieurs thèmes, créer le ou les personnages, le monde, etc... Il y a aussi des softs dont l'idée tient très bien en une ou deux phrases. Par exemple, les jeux de courses ne demandent pas un abondant synopsis. Par contre, il faudra faire un choix selon que les designers veulent des circuits réalistes (comme dans *Fl Pole Position*), ou alors des circuits complètement fous comme dans *Extrême G*.



Il y a autant de scénarii que de styles de jeux. Mais la complexité des histoires n'a rien à voir avec la complexité du développement du soft. Une fois l'idée et le principe de base établis, il faut maintenant passer de l'abstrait au concret. Pour cela, le plus simple est bien évidemment de tout mettre sur une feuille de papier.

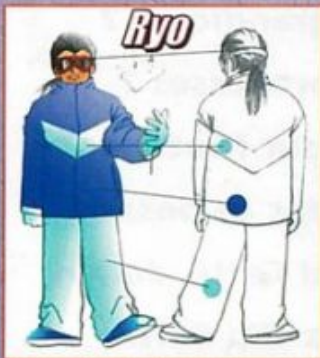
On fait donc un "story board".

Que ce soit pour créer un jeu de plates-formes, d'aventure, de rôle ou de courses, les développeurs sont unanimes :

le story board est ce qu'il y a de plus facile pour commencer à concrétiser leur idée.

A ce stade, les idées prennent forme et chacun peu déjà s'imaginer le jeu ainsi que qu'il contiendra. Les "story boards" peuvent être de simples dessins, ou une véritable B.D. pour les jeux aux scénarii très complexes. Il existe des cas bien évidemment, où tout est déjà pré-défini : lorsqu'une société décide de faire un jeu à partir d'un film ou d'un dessin animé, le personnage et l'histoire sont déjà créés. Par exemple, le dessin animé *Hercules* a été repris en jeu vidéo.

Après cette première étape commence le travail des infographistes qui, à partir de ces dessins, vont modéliser les formes et les volumes en 3D.



Pour faciliter leur travail, les dessinateurs donneront à ceux-ci des effets de perspective, des zones d'ombres, etc... tout ceci va permettre aux infographistes de mieux approfondir les formes et d'avoir une meilleure représentation mentale du jeu. Tous ces éléments vont être exportés et convertis au format utile pour la programmation.



C'est ensuite que commence le travail des programmeurs.



Pour *GoldenEye*, les infographistes ont créé un modèle en 3D à partir de photos.



MarKEtING : une bONne politiQUE est EssENTielle

Elle est bien révolue l'époque où les programmeurs développaient tranquillement leur jeu, enfermés des heures durant au fond de leur cave ou dans leur garage. Ils avaient le temps, pas de contraintes financières, créaient des jeux pour eux-mêmes, sans se soucier de l'impact médiatique. Tout restait à inventer, le secteur promettait un bel avenir. Aujourd'hui, les jeux vidéo sont une grosse industrie, une machine à faire des sous. Ils sont plus devenus un monde de businessmen que de passionnés.

Actuellement, pour sortir un jeu, il faut avoir de l'argent, beaucoup d'argent.



D'ailleurs, il est souvent très difficile de savoir combien coûtera le développement. Surtout lorsqu'il s'agit de Nintendo. En effet, à chacune des étapes du développement d'un jeu d'un éditeur extérieur, la société nippone procède à une phase de vérification. Si cela ne convient pas, on recommence, jusqu'à ce que le jeu satisfasse aux conditions requises par Nintendo. Ainsi, une sortie peut être repoussée pendant quelques temps... et le temps, c'est de l'argent. Il faut aussi penser aux royalties que touchera Nintendo, et d'autres sociétés qui ont participé au projet... bref, il faut réfléchir à tout, aux investissements, coûts de fabrication, packaging, publicité (télévisée et dans la presse), salaires, etc... Eh oui, il ne faut pas croire que tout se fait seulement au cours des derniers mois de développement du jeu.

Car n'oublions pas que les titres sortis à Noël sont, pour la plupart, des produits qui ont été développés depuis un ou deux ans !!

Il faut donc penser très tôt à préparer le terrain. Et les coût de développement d'un jeu à l'heure actuelle sont compris entre 10 et 20 millions de francs. Autant dire qu'il faut en vendre des cartouches !!! Mais certaines sociétés, comme Ubi Soft, arrivent à amortir ce coût en réutilisant un moteur préalablement créé pour développer un jeu déjà lancé. C'est ainsi que le moteur développé pour Pod a été réutilisé pour créer F1 Racing ; le coût du développement a baissé de 20%. Pour Tonic Trouble, les programmeurs espèrent utiliser leur nouveau moteur "Open Space", dans le but de programmer un futur Rayman 3D.



Amortir les coûts de production, voilà le vrai secret d'une bonne rentabilité !!

Bref, le marketing est sans doute l'élément essentiel pour bien faire connaître et vendre un jeu. Il faut être présent partout. Au développement, à la mise en vente du produit, sur les divers supports média, etc... Sans oublier que pour Nintendo, il faut penser aux royalties... On n'a rien sans rien !

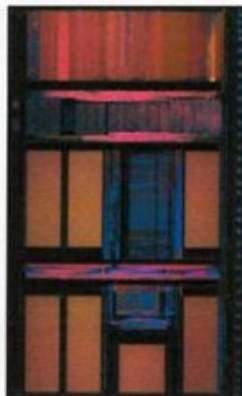
daNS les EntRAilles dE la bête

La Nintendo est la console la plus puissante du marché, grâce à ses composants issus de la technologie Silicon. Pour votre information, sachez que ces composants sont présents dans les "stations de travail" Silicon Graphics. Avant de savoir programmer un jeu, il est bien souvent indispensable de connaître le hardware d'une console. Et celui de la Nintendo est assez impressionnant.

Prêts à ouvrir la caverne d'Ali Baba ?!

Le coprocesseur "Reality". Ce chip contrôle les graphismes, le son, les effets de mip-mapping, d'anti-aliasing, etc... Il fonctionne à 62,5MHz. Bref, c'est ici que se situent toutes les données qui permettent d'effectuer les "effets spéciaux".

Voici le fameux processeur central Mips R4300i Risc 64 bits. Il tourne à 93,75MHz. C'est le coeur de la Nintendo. C'est par lui que passent toutes les informations. Il contrôle et coordonne l'ensemble des fonctions des autres co-processeurs. Bref, c'est le Big Boss, quoi !!



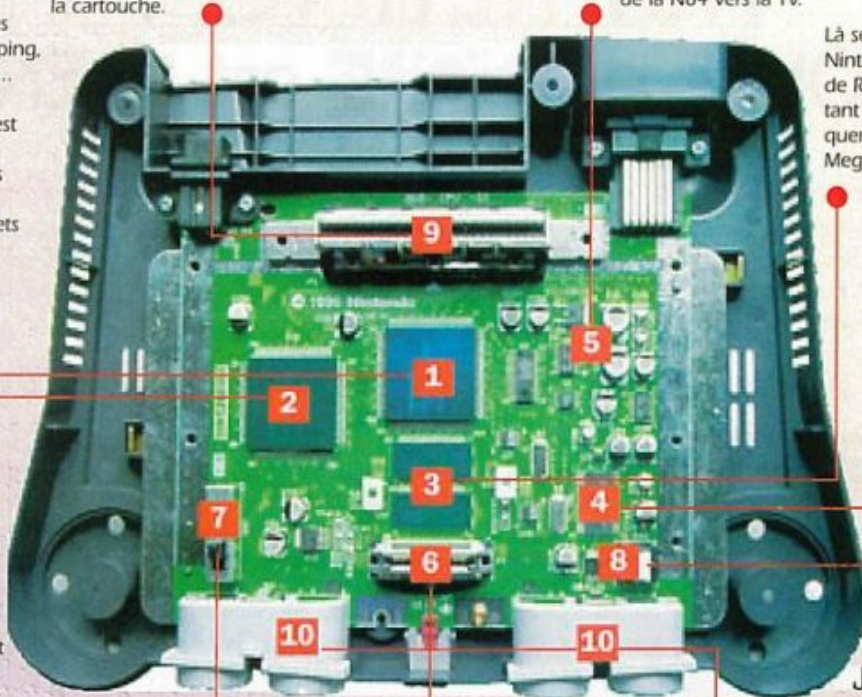
Voici deux représentations du processeur central Mips R4300i Risc produit par NEC CORPORATION et conçu par MIPS et SILICON GRAPHICS.

Port cartouche où se connecte... la cartouche.

Résistances de conversion. Ces "resistors" sont utilisées pour la vidéo, l'audio, et les graphismes. Ils aident dans la « conversation » des signaux audio/video de la N64 vers la TV.

Là se trouve la RAM de la Nintendo. 4 Méga octets de RAM et un bus permettant de transférer (théoriquement) jusqu'à 563 Megabits/seconde.

Le processeur de gestion des périphériques permet tout simplement la communication entre la N64 et les divers périphériques utilisés : manettes, DD64, modem...



Bouton marche/arrêt : là, je pense qu'il n'y a pas besoin de commentaires.

Baie d'extension pour la Ram : slot réservé pour une future carte d'extension RAM. On ne sait pas très bien, encore, comment cela se présentera, ni si Nintendo espère réellement s'en servir. Surtout ne la retirez pas !

Bouton de reset : les autres consoles disposent d'un reset "hard" très violent, c'est-à-dire qui coupe et remet le courant. Celui de la Nintendo est un reset "soft" micro-programmé qui avertit la cartouche du reset.

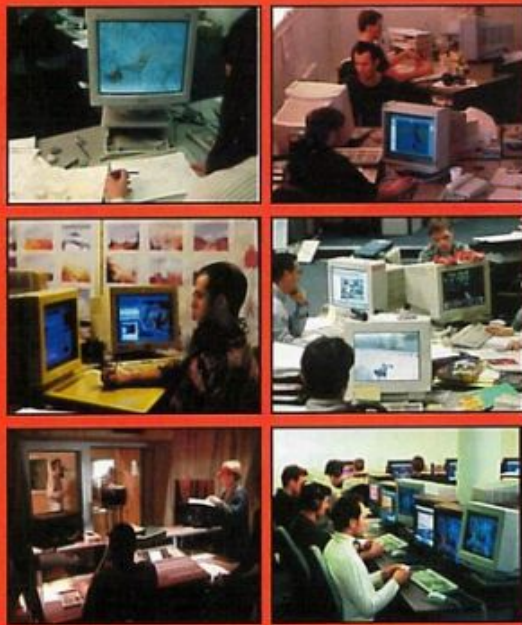
Port manette

ProgrAMmatiOn : uNE étAPe difficile

Après avoir créé le background, l'histoire, le personnage et tout ce qui va faire vivre le jeu (sans oublier qu'au même moment, le marketing travaille déjà sur la façon de commercialiser le produit), les programmeurs commencent alors leur tâche.

En fait, ils ont déjà commencé leur travail depuis des semaines, voire des mois. Passer d'une machine à une autre demande beaucoup de temps. Il faut l'étudier, connaître ses capacités, ses librairies, etc... Pas question de brancher le kit de développement sur le PC et de commencer à écrire les données !!

Là encore, si le principe du développement d'un jeu sur N64 ne diffère pas trop par rapport à celui d'une autre console (il faut "simplement" connaître et savoir utiliser le langage "C"), il existe quelques spécificités que l'on ne trouve pas ailleurs. D'une part, la programmation ne



Le monde des jeux vidéo est devenu une vraie industrie qui embauche énormément de monde (tous passionnés de jeux vidéo ?...). Dans ces locaux, chacun s'occupe d'une étape différente d'un jeu. Les moyens mis en oeuvre sont souvent énormes, et on se croirait vraiment dans un studio de dessin animé Disney III

s'effectue pas de la même façon sur cartouche ou sur C.D. D'autre part, la puissance de la Nintendo permet de faire plus de choses (3D libre, anti-aliasing, correction de perspective,...) que sur Playstation ou sur Saturn.

Nintendo est très exigeant sur la qualité de ses jeux. Ainsi, existe-t-il un **cahier des charges** qui doit être scrupuleusement suivi par l'éditeur-tiers lors de la programmation s'il ne veut pas retrouver son jeu dans un vieux tiroir. Ce cahier intervient lorsqu'il s'agit de la sauvegarde, du rumble-pack, de la jouabilité, etc...car, bien évidemment, les programmeurs ne peuvent pas gérer cela personnellement. Ils sont obligés de respecter certains critères pour que demeure une certaine cohérence entre les différents jeux Nintendo.

Le point sur lequel Nintendo est sans doute intraitable, c'est la jouabilité.

C'est la raison pour laquelle le jeu, après être passé entre les mains des testeurs de la société «tiers», est soumis au contrôle inévitable des testeurs de Nintendo. Et cette étape de vérification peut durer longtemps...

Chaque console est spécifique et dispose de données différentes, comme les bibliothèques, sortes de codes liés à la console. S'il n'est pas aisé de programmer sur Saturn, c'est en partie parce que les bibliothèques sont assez difficiles à utiliser. Les programmeurs les emploieront en passant par certaines fonctions afin de s'assurer que, dans le cas où un nouvel élément apparaît (memory pack, par exemple), le code est toujours compréhensible.

Pour savoir programmer sur la Nintendo 64, il faut pouvoir utiliser le langage C et savoir

aussi manier l'API de la Nintendo.

L'API est l'ensemble des fonctionnalités associées à une machine. C'est-à-dire que la Nintendo a une série de fonctions que le programmeur peut utiliser : affichage, traitement de certaines tâches, etc... Quand le programmeur connaît ces fonctions et sait utiliser le langage C, il peut réunir les deux pour en faire un code intelligible par la Nintendo. Le plus difficile, c'est bien évidemment de comprendre comment fonctionne une Nintendo, c'est-à-dire quand et comment envoyer l'ordre d'exécuter une tâche à un processeur, pour pouvoir optimiser au maximum le hardware de celle-ci. Bref, tout le secret est de savoir jongler avec les fonctions de la Nintendo.

Les bibliothèques de la N64 sont plus puissantes que celles des autres consoles, ce qui permet une plus grande vitesse de programmation et des jeux au design très élaboré (comme Mario 64 qui dispose d'une 3D libre).

Mais, et cela peut paraître contradictoire, la Nintendo est très difficile à programmer, parce qu'elle dispose de peu de mémoire par rapport à une Playstation. Le support (cartouche) lui-même est très contraignant car il ne dispose pas d'autant de mémoire qu'un CD-Rom. Mais encore faut-il relativiser la chose. Malgré ses défauts, le support cartouche est très pratique : il fait gagner un temps fou lors des chargements des programmes, alors que sur C.D., les temps d'ac-

Des formations au sein des entreprises ?

Au sein de la société Ubi Soft, il n'existe aucune formation pour apprendre à programmer sur une nouvelle console. Les programmeurs reçoivent un kit de développement de la part de Nintendo, et c'est à eux de s'y plonger. Mais, à peu près chaque année, Nintendo organise un colloque où se réunissent des programmeurs afin d'y découvrir les dernières données informatiques, ainsi que quelques conseils pour optimiser le hardware, les nouvelles bibliothèques, etc...

En fait, le programmeur est en formation de façon continue. A chaque fois, il découvre de nouvelles techniques de programmation, et de nouveaux domaines restés jusque là insoupçonnés. Bref, il découvre la machine tout en travaillant dessus. Rien n'est gagné d'avance, tout reste à découvrir. A l'époque du lancement de la Super Nintendo, qui aurait pu imaginer que quelques années plus tard il serait possible de faire tourner des jeux en 3D comme Donkey Kong Country, Yoshi's Island ou Killer Instinct ?... On voit bien que personne ne peut véritablement dire de quoi est capable une console, ni où sont ses limites. Peut-être dans quelques années la Nintendo sera-t-elle capable d'afficher des "nurbs", ou encore en mesure d'utiliser la technique du "ray-tracing" ?

cés sont extrêmement longs. Bien évidemment, lorsqu'il faut importer beaucoup de données, cela demande au total un temps assez important. C'est beaucoup plus rapide avec la cartouche : 50 megas/seconde. Bref, grâce au support cartouche, on a un chargement qui s'effectue presque en temps réel.

Le kit de développement

C'est une console Nintendo avec une interface.

Celle-ci permet de programmer la console à partir d'un PC (ou d'une Silicon Graphics) grâce à des outils de développement spécifiques, en particulier un compilateur et un débogueur (qui sont des programmes PC. donnés par Nintendo).

Le kit comprend donc une console modifiée, une cartouche réinscriptible connectée à un P.C. et une carte informatique Nintendo remplie de programmes.

Le programme d'un jeu se fait donc sur PC. Un code, qui utilise la librairie Nintendo, est écrit au préalable. A un certain moment, on utilise le compilateur de façon à transformer ce code en le rendant intelligible pour la Nintendo. On dit alors que l'on compile les données. Par l'intermédiaire du kit de développement, on envoie cet "exécutable" dans la Nintendo. Lorsque qu'on allume la console, ce code est exécuté. Le débogueur, lui, permet de trouver les bugs dans le code, en exécutant un autre programme pas à pas. Enfin, sachez que la mémoire d'un kit de développement est réinscriptible à l'infini et qu'elle possède une capacité de 256 mégabytes, alors qu'une cartouche normale n'en contient, au maximum, que 128. Je n'ai, hélas, pas pu prendre de photo du kit Nintendo, étant donné que l'on ne m'en a pas donné l'autorisation. Un secret jalousement gardé !

Le 64DD : déjà des jeux en préparation !!

Certains éditeurs comme Ubi Soft, ont déjà reçu le kit de développement pour le 64DD.

Ce qui veut dire que :

- le 64DD est vraiment prévu pour l'Europe
- ce n'est plus une attente utopique
- préparez-vous à jouer à des jeux qui risquent d'être formidables.

Le kit du 64DD ressemble à un 64DD mais qui possède un disque optique spécial. On branche ce kit 64DD sous le kit N64. Le programme passe d'abord par la cartouche puis est envoyé dans le 64DD. On inscrit tout ça sur le disque optique puis, comme pour un C.D, on le duplique en plusieurs centaines de fois sur d'autres disques optiques.

Je n'ai pas pu prendre de photos...

Mais si vous aviez vu mon état en rentrant à la redac !! J'ai pu le voir (et sûrement suis-je l'un des rares à avoir eu ce "privilège" ?!) et en plus le toucher ! Vivement sa sortie officielle !



La cartouche

Voici comment se présente une cartouche (Eprom). Elle est tout simplement constituée de barrettes de ROM. C'est là où sont stockées toutes les données d'un jeu. Bien entendu, la cartouche ne peut utiliser ces données toute seule. Ces barrettes à leur extrémité sont équipées de fiches qui, lorsque vous connectez l'éprom au port de la Nintendo, transmettent, via le BUS, les données vers la RAM de la console. Il n'y a donc pas de chargement car vous connectez presque directement la ROM sur le hardware de la Nintendo. Ainsi, les informations sont véhiculées et transférées très rapidement.

De plus, la cartouche est plus fiable et plus robuste. En outre, elle coûte cher à fabriquer, et c'est Nintendo qui se charge de la duplication.

Enfin, il existe des cartouches de différentes tailles de mémoire. En général, elles font 96 Megabytes (c'est à dire 12MO, 1 octet = 8 bytes). Zelda devrait faire normalement 256 Megabytes, donc devrait être contenu dans une cartouche de 32 MO. Plus elle contient de MO, plus la cartouche coûte cher, puisqu'elle contient plus de plaquettes de ROM.



Everything about

64DD

Voici le disque optique. Par rapport à la cartouche, il a une capacité de stockage de 512 mégabytes (64 MO).



Le gLOSSaiRE

Vous trouverez dans ce glossaire la définition des principaux termes utilisés dans notre dossier. J'en ai aussi ajouté d'autres qui, même s'ils n'en font pas partie, sont généralement rencontrés dans le monde de la programmation.

ABI (Audio Binary Interface) : portion du programme de la librairie de la Nintendo 64 utilisée pour le microcode de la musique et des effets sonores.

Anti-aliasing : technique permettant d'interpoler les couleurs de deux éléments graphiques à leur frontière. Existe sur N64, mais pas sur Playstation. L'anti-aliasing est implanté dans le coprocesseur "Reality".

Burning : terme utilisé pour la duplication des cartouches. On "grille" les résistors de l'Eprom, c'est-à-dire qu'on les libère de leurs programmes.

CPU (Central Processing Unit) : c'est le processeur principal de la console. Le CPU de la N64 est un processeur de la série MIPS 400 64 bits qui "tourne" à 93,75 Mhz.

Texture mapping à correction de perspective : une technique utilisée pour réaliser des textures réalistes, tout en tenant compte de la perspective. La correction de perspective est implantée dans le coprocesseur "Reality".

Double buffering : zone mémoire réservée pour composer les images à envoyer au joueur. Les programmeurs travaillent donc souvent avec deux zones : une que l'on envoie au joueur, et l'autre que l'on prépare pour le tour d'après. En effectuant un "roulement" à l'aide de ces deux zones, le joueur voit sans cesse une nouvelle image.

Environment mapping : une technique qui permet de créer des reflets très réalistes sur une surface. Existe seulement dans le hardware de la Nintendo. Exemple : le reflet du soleil sur l'eau.

GBI (Graphics Binary Interface) : partie de programme de la librairie de la N64 pour utiliser le microcode graphique.

Gouraud Shading : technique utilisée sur les polygones pour donner des effets d'ombre. Exemple : on peut faire varier une lumière et on calcule en temps réel son influence sur les personnages ou les objets statiques.

Hardware : terme qui regroupe les éléments (processeurs, co-processeurs, etc..) et les fonctions internes d'une console.

Lumière spéculaire : technique dérivée du "specular reflection mapping". Permet des rendus métalliques sur les objets. Exemple : Mario qui se transforme en métal dans Mario 64. La Lumière spéculaire est implantée dans le coprocesseur "Reality".

Mhz : ordre de grandeur qui sert à mesurer le nombre d'instructions exécutées par le processeur. Par exemple, le CPU

réalise environ 100 millions d'instructions par seconde.

Microcode : programme qui régit les fonctions du "Reality Signal Processor". Ces microcodes sont utilisés pour tout ce qui est du domaine audio. Mais il en existe d'autres pour les graphismes.

Mip mapping : technique de "texture mapping" plus évoluée, permettant d'éliminer tous les phénomènes de pixellisation. Existe sur N64, mais pas sur Playstation.

MIPS : Millions d'Instructions Par Seconde. Une unité de mesure pour définir la capacité de travail d'un processeur. Par exemple, le CPU de la N64 est à 125 MIPS.

Multi processing : système permettant l'exploitation de plusieurs processeurs. Par exemple, la Nintendo 64 peut se servir séparément du CPU, du Reality Signal Processor, du Reality Display Processor.

Multitasking : système d'opérations consistant à faire travailler deux ou trois processeurs simultanément. Ce système est implanté dans le coprocesseur Reality.

Nurbs : les nurbs sont le futur des polygones. Au lieu d'être des objets plats (triangles), ce seront des objets courbes. Exemple : au lieu d'avoir des personnages modélisés en 3D avec des épaules cubiques, on aura des épaules et des formes beaucoup plus arrondies.

Polygones : généralement des triangles, éléments à la base de tous les objets affichés en 3D (Mario 64, Tonic Trouble, F1 Pole Position, Extreme G, etc...).

RAM (Random Access Memory) : c'est la mémoire vive de la Nintendo 64. Il s'agit de la mémoire de travail d'un processeur ; c'est là où il range, lit et effectue les données transmises. Notez que la N64 a besoin de moins de RAM que les autres consoles. En effet, les ROM des cartouches servent déjà de stockage.

Ray tracing : le ray tracing est une technique d'affichage déjà utilisée par l'image de synthèse. Le jeu vidéo ne l'utilise pas encore, car cela demande des machines ultra puissantes. Mais, d'ici trois ou quatre ans, on commencera à jouer avec des jeux en images de synthèse.

RCP (Reality CoProcessor) : le coeur de la Nintendo 64. Ce processeur exécute toutes les informations graphiques et audio.

RDP (Reality Display Processor) : ce processeur est encastré à l'intérieur du coprocesseur "reality" et exécute toutes les

Sommaire

Sorties Jeux

News

Enquête

Savoir

Labo

opérations concernant la "texture mapping" et l'"anti-aliasing".

RDRAM (Rambus Dynamic Ram) : développé par Rambus INC., le RDRAM permet d'accélérer l'accès à la RAM.

ROM (Read Only Memory) : c'est la mémoire morte. Sans RAM, les informations qu'elle contient ne peuvent être utilisées. La ROM est contenue dans la cartouche.

Sprite(s) : ancêtre du polygone ; élément(s) dont on se servait pour faire des jeux 2D.

Texture mapping : appliquer des motifs, des dessins sur un ou plusieurs polygones en même temps. Exemple : sur les polygones formant un mur, on applique une texture de

brique, ou de bois pour lui donner de la matière.

TLM (Trilinear MIP Map Interpolation) : une version combinant la "Trilinear Interpolation" avec le "Mip-Mapping". Généralement considéré comme le must pour le rendu des textures.

Trilinear interpolation : technique qui permet d'améliorer l'apparence des textures selon la distance grâce à un mélange de leurs couleurs.

Z buffer : zone mémoire réservée pour effectuer un positionnement quasi parfait des objets 3D à afficher. Permet de mieux contrôler l'intersection des polygones. N'existe pas sur Playstation.

Quelques adresses utiles

Si vous voulez devenir infographiste,

je vous conseille de téléphoner, ou bien d'aller directement à la S.E.M.A. (Société d'Encouragement aux Métiers d'Art). Vous aurez accès à une documentation gratuite.

S.E.M.A.
252 rue du fbg Saint-Honoré
75008 Paris
Tél : 01-55-78-86-08

D'autres adresses utiles :

Ecole Nationale des Arts Appliqués
63 rue Olivier de Serres
75015 Paris
Tél : 01 53 68 16 90.

Attention, il n'existe pas de "classe" spécialisée en infographie dans cette école. Ici, c'est de l'Art pur et dur, mais certains cours proposent une matière (en option ou en filière générale) d'infographie. Il faut avoir le niveau bac (bac+1, c'est encore mieux !!). Vous passerez bien évidemment un concours (le 27 avril à 14 heures) et bénéficierez d'une remise à niveau en Art si vous n'y connaissez rien !!

Ecole des métiers de l'image
Centre Gobelins
73 bd Saint-Marcel
75013 Paris
Tél : 01 40 79 92 50

Il faut avoir le niveau bac. Cette école propose une spécialité pour devenir véritablement infographiste. On vous apprend en fait à devenir "assistant réalisateur multimédia". Vous avez un an d'études en alternance.

Pour devenir programmeur :

I.U.T. de Reims
rue des Crayères, B.P. 1035
51687 Reims, Cedex 2
Tél : 03 26 05 30 50

Il existe dans cette école un diplôme d'informaticien programmeur, reconnu par l'état. Pour y entrer, il faut posséder un Bac C. Les études durent 2 ans. Pour les Bac+2, on vous proposera une année spéciale : on y apprend les mêmes choses, mais le cycle dure un an.

Université de Paris-VIII
Formation "arts technologiques de l'image".
93256 St-Denis Cedex 02
Tél : 01 49 40 66 04

Pour y entrer il faut avoir un Bac+2, et on n'est définitivement accepté que sur... concours !! Il vaut mieux déjà avoir un bon dossier artistique. Dans cette école, on vous apprend à créer votre programme, votre logiciel. Cela dure deux ans (un an de licence, un an de maîtrise). Le concours débute le 15 juin.

Cela dit, ne vous leurrez pas. Même si vous avez tous les diplômes nécessaires, les éditeurs ne vous prendront pas forcément au sein de leur entreprise. Les conditions requises sont d'ordre beaucoup plus personnel. Il faut être créatif, posséder une excellente culture vidéo-ludique, et surtout avoir déjà programmé des jeux (sur Atari ST par exemple). Vous voyez, la partie n'est pas gagnée !!

