INTRODUCTION AUX JEUX VIDEO

Chapitre 03 : Les outils de création de jeux vidéo

L'objectif de ce chapitre est de montrer aux apprenants les différents outils et technologies utilisés dans le processus de création d'un jeu vidéo.

A la fin de cette partie, l'apprenant sera capable :

1. de connaitre les outils et les technologies pour la conception de jeu vidéo
2. de choisir les bon outils pour son projet de création de jeu vidéo.

Mr Maodo KANE

Enseignant-chercheur en Informatique

Université Gaston Berger de Saint-Louis du Sénégal

UFR des Civilisations, Religions, Arts et Communication (CRAC)

maodo.kane@ugb.edu.sn

Table des matières

III. Les outils de création de jeux vidéo 3
III.1 Les outils de programmation 3
III.1.1 Définition de la programmation de jeux vidéo 3
III.1.2 Le langage de programmation 4
III.1.3 Les interfaces de programmation applicative (API) 5
III.1.4 Les environnements de développement intégré (IDE) 8
III.2 Les moteurs de jeu 11
III.2.1 Moteur de jeu pour débutant 12
III.2.2 Moteur de jeu pour les experts 14
III.3 Les outils d'infographie 17
III.3.1 Logiciels 2D 17
III.3.2 Logiciels 3D 19
III.3.3 UML en jeu vidéo 23
Sources 26

III. Les outils de création de jeux vidéo

III.1 Les outils de programmation

III.1.1 Définition de la programmation de jeux vidéo

La programmation de jeux vidéo est un processus entrant dans la création d'un jeu vidéo. Ce processus est effectué par un développeur de jeux vidéo via la programmation informatique.

Fondamentalement, la programmation d'un jeu vidéo se réduit à répéter à intervalles de temps réguliers les actions suivantes :

- gérer les interactions d'un agent avec l'environnement en fonction des entrées de l'utilisateur,
- gérer les phénomènes non directement contrôlés par l'utilisateur,
- créer une image à partir de ces informations et l'afficher à l'écran (de même pour le son).

La gestion des interactions consiste à prévoir les actions possibles sur l'environnement, de détecter leur déclenchement et de gérer des réponses. Par exemple, pour créer une interaction « ouvrir la porte », il faut décider des processus qui permettront son déclenchement (ex: le joueur doit appuyer sur le bouton prévu à cet effet, le personnage doit être assez proche de la porte et il faut qu'il lui fasse plus ou moins face), ensuite il faut, durant le jeu, détecter que ces conditions sont remplies. Si c'est le cas, alors une réaction est générée, dans notre exemple, afficher sur l'écran une porte qui s'ouvre.

Les phénomènes non directement contrôlés par l'utilisateur sont multiples et varient beaucoup en fonction du jeu. Il peut s'agir de gérer l'intelligence artificielle des agents non contrôlés par l'utilisateur, gérer les collisions entre agents et la physique (pour les jeux qui utilisent un modèle physique réaliste).

Ces deux premières parties font partie du gameplay (c'est-à-dire, des règles d'interaction) d'un jeu qui est normalement décidé par un game designer et implémenté par le programmeur.

La création de l'image et son affichage devient de plus en plus complexe à mesure que le niveau graphique des jeux s'élève. L'image est construite petit a petit dans un espace mémoire (le framebuffer) et n'est envoyé à l'écran qu'une fois le dessin fini dans son ensemble.

La difficulté est donc de pouvoir créer une image de bonne qualité dans un laps de temps très court, tout en gérant les différents éléments du jeu. C'est dans cette optique qu'aujourd'hui la programmation de jeux 3D est de plus en plus liée au matériel informatique, en utilisant au mieux les fonctionnalités d'accélération matérielle des cartes graphiques afin d'accélérer au maximum le traitement des objets 3D, et en optimisant le code source.

La fréquence de répétition de ces opérations varie fortement en fonction de la plate-forme (ordinateur, console, téléphone portable, etc.). On considère qu'il faut au moins 24 images par seconde pour donner une bonne impression de mouvement au cinéma mais il faut une fréquence plus élevée dans les jeux vidéo. Toutes les parties d'un programme n'ont pas forcément besoin d'être traitées à chaque image (sauf la création et l'affichage de cette image).

Les programmeurs de jeux vidéo sont aussi amenés à développer divers logiciels et outils informatiques utilisés dans la chaîne de production du jeu vidéo ; outils destinés par exemple aux autres acteurs du développement (comme les infographistes, animateurs, game designers, ...) ou encore à exécuter du traitement par lots.

III.1.2 Le langage de programmation

Le développement d'un jeu vidéo utilise plusieurs langages de programmation. Les premiers jeux étaient développés en langage assembleur, version humaine du langage machine, puis, permettant un développement plus rapide, vers des langages plus structuré lors de leurs apparitions. Le langage assembleur est cependant toujours utilisé pour les tâches nécessitant une très grande optimisation. Sur les premiers micro-ordinateurs, le BASIC était le langage le plus répandu, puis, il a été progressivement supplanté par le langage C, venant des systèmes Unix et permettant une approche plus bas niveau.

Enfin, il existe aujourd'hui de nombreux moteurs de jeux, qui s'occupent de ces couches bas niveau. Ceux-ci peuvent être des API utilisés dans des langages de programmation (C, C++, etc), ou langage de script (JavaScript, lua, Python, Ruby, etc.)

Le premier choix à faire est le langage. En théorie, il est possible de prendre n'importe quel langage pour faire son jeu. En pratique, certains langages seront plus adaptés que d'autres.

Il n'y a pas de langage parfait. Chacun a son point de vue par rapport aux langages, tiré de sa propre expérience.

Certaines plateformes ne supportent pas tous les langages. Ainsi, si vous souhaitez faire un jeu pour des plateformes autres que le PC, renseignez-vous des contraintes pour cette plateforme.

Comme première indication, l'assembleur est fortement déconseillé. Le langage est trop bas niveau pour pouvoir faire des jeux simplement. Seules les vieilles machines (Game Boy, Amiga, Atari) nécessitent encore l'utilisation de ce langage.

Souvent les débutants pensent à utiliser le **C** car c'est un des premiers langages appris à l'école. De plus, beaucoup de bibliothèques pour programmer des jeux vidéo sont disponibles dans ce langage, toutefois, le C demande une grande rigueur. Pour se faciliter la vie, il est préférable de commencer par un langage plus simple afin de s'alléger de la gestion de la mémoire qui, dans le C, est manuelle.

Le **C++** est le langage par excellence pour les jeux vidéo. Il est rapide comme le C et apporte les outils pour la programmation orientée objet. De plus, le C++ permet de minimiser l'usage des pointeurs et de ne pas à avoir à gérer la mémoire manuellement. Les bibliothèques du C sont compatibles avec le C++ même s'il sera toujours préférable d'utiliser les bibliothèques C++. Toutefois, le langage demande beaucoup de rigueur.

Le **Java** est un langage de choix, car couramment utilisé tout en étant simple. De plus, ce langage est privilégié si vous voulez faire un jeu pour Android.

Le **C#** est semblable au Java. Toutefois, il a souvent été limité aux plateformes Microsoft. Avec, vous pouvez développer des jeux avec Monogame. Finalement, ce langage est utilisé dans les moteurs **Unity 3D** et **NeoAxis**. Ce langage est fortement utilisé chez les professionnels du jeu vidéo pour la création des outils liés à la conception du jeu (éditeurs, logiciels utilitaires, etc.).

Le **BASIC**, popularisé car implémenté d'office sur les vieux ordinateurs, est un langage assez simple. Il est toujours possible de faire des jeux avec en utilisant des variantes telles que **PureBasic** ou **Quick Basic**, mais ces solutions ont vieilli.

Le **Python** est un langage de script simple et puissant. Il est généralement facile à apprendre et à utiliser. De plus, il existe de nombreuses bibliothèques aidant à la programmation d'un jeu, comme la très célèbre **Pygame**.

HTML et **CSS** vous permettent de construire et de mettre en forme l'interface utilisateur de votre jeu web. Par exemple, l'élément HTML **<canvas>** permet de gérer des graphisme en deux dimensions.

JavaScript, est un langage de programmation utilisé sur internet. Il a d'excellentes performances sur les navigateurs modernes et est sans cesse amélioré. Utilisez sa puissance pour coder vos jeux, ou regardez son utilisation dans des technologies comme **Emscripten** ou **Asm.js** pour porter facilement vos jeux existants.

III.1.3 Les interfaces de programmation applicative (API)

En informatique, une interface de programmation applicative (souvent désignée par le terme API pour application programming interface) est un ensemble normalisé de classes, de méthodes ou de fonctions qui sert de façade par laquelle un logiciel offre des services à d'autres logiciels. Elle est offerte par une bibliothèque logicielle ou un service web, le plus souvent accompagnée d'une description qui spécifie comment des programmes consommateurs peuvent se servir des fonctionnalités du programme fournisseur.

Dans l'industrie contemporaine du logiciel, les applications informatiques se servent de nombreuses interfaces de programmation, la programmation se fait en réutilisant des briques de fonctionnalités fournies par des logiciels tiers. Cette construction par assemblage nécessite pour le programmeur de connaître la manière d'interagir avec les autres logiciels, qui dépend de leur interface de programmation. Le programmeur n'a pas besoin de connaître les détails de la logique interne du logiciel tiers, et celle-ci n'est généralement pas documentée par le fournisseur.

Des logiciels tels que les systèmes d'exploitation, les systèmes de gestion de base de données, les langages de programmation, ou les serveurs d'applications comportent une interface de programmation.

Voici une liste d'API très utilisés en jeu vidéo

OpenGL

OpenGL (Open Graphics Library) est un ensemble normalisé de fonctions de calcul d'images 2D ou 3D lancé par Silicon Graphics en 1992. Cette interface de programmation est disponible sur de nombreuses plateformes où elle est utilisée pour des applications qui vont du jeu vidéo jusqu'à la Conception Assistée par Ordinateur (CAO) en passant par la modélisation.

OpenGL permet à un programme de déclarer la géométrie d'objets sous forme de points, de vecteurs, de polygones, de bitmaps et de textures. OpenGL effectue ensuite des calculs de projection en vue de déterminer l'image à l'écran, en tenant compte de la distance, de l'orientation, des ombres, de la transparence et du cadrage.

L'interface regroupe environ 250 fonctions différentes qui peuvent être utilisées pour afficher des scènes tridimensionnelles complexes à partir de simples primitives géométriques. Du fait de son ouverture, de sa souplesse d'utilisation et de sa disponibilité sur toutes les plates-formes, elle est utilisée par la majorité des applications scientifiques, industrielles ou artistiques 3D et certaines applications 2D vectorielles. Cette bibliothèque est également utilisée dans l'industrie du jeu vidéo où elle est souvent en rivalité avec la bibliothèque de Microsoft : Direct3D. Une version nommée OpenGL ES a été conçue spécifiquement pour les applications embarquées (téléphones portables, agenda de poche, consoles de jeux…).

Plusieurs bibliothèques sont développées à partir d'OpenGL afin d'apporter des fonctionnalités qui ne sont pas disponibles dans la bibliothèque OpenGL elle-même :

- **GLU** (OpenGL Utility Library) : apporte les fonctionnalités comme la gestion de la matrice de projection pour mettre en place une vue orthogonale ou en perspective, la gestion de la matrice de visualisation avec une routine de type « caméra », le redimensionnement d'images, la construction automatique de Mipmaps, la fragmentation de polygones, la gestion d'objets quadriques (sphère, cylindre, disque), la gestion des courbes et des surfaces de Bézier, la gestion des surfaces NURBS (Non Uniform Rational B-Spline), la Vérification de la présence d'extensions OpenGL, etc.

- **GLUT** (GL Utility Toolkit : combine l'accès à des fenêtres graphiques OpenGL et, de façon unifiée, aux interfaces utilisateur du système d'exploitation, quel que soit celui-ci);

- **GLUI** (OpenGL User Interface Library) : apporte le support de divers éléments de contrôle tels que les boutons, les cases à cocher, les zones de texte éditables et statiques, les listes déroulantes, etc.

- **GLEW** (OpenGL extension wrangler library) : apporte une solution à la complexité de la gestion des extensions d'OpenGL. À l'exécution, elle détecte les extensions supportées par la machine hôte et initialise les fonctions associées à ces extensions.

- **GLFW** (GL Frame Work) : offre un jeu de routines pour la gestion des fenêtres OpenGL, proposée en alternative à GLUT

- **GLM** (OpenGL Mathematics) : apporte au programmeur C++ tout un ensemble de classes et de fonctions permettant de manipuler les données pour OpenGL.

DirectX

Microsoft DirectX est une collection de bibliothèques destinées à la programmation d'applications multimédia, plus particulièrement de jeux ou de programmes faisant intervenir de la vidéo, sur les plates-formes Microsoft (Xbox, systèmes d'exploitation Windows).

Les composants constituant DirectX sont :

- **DirectDraw** : destiné aux graphismes en deux dimensions. Cette partie n'est plus mise à jour mais reste utilisée par un grand nombre de jeux. Cette bibliothèque est considérée comme obsolète (deprecated), remplacée par Direct2D.

- **Direct2D** : remplace, en plus complet, DirectDraw. Disponible pour Windows 7, et pour Vista sous forme de mise à jour.

- **DirectWrite** : pour les polices. Offre notamment un anticrénelage bidirectionnel. Disponible pour Windows 7, et pour Vista sous forme de mise à jour.

- **Direct3D (D3D)** : Pour les graphismes 3D.

- **DXGI** : Pour énumérer les cartes et les écrans et gérer les swap chains pour DirectX 10 et au-dessus.

- **DirectInput** : permet de gérer les entrées utilisateur de nombreux matériels comme les claviers, souris et joysticks ou tout autre matériel connectable à un PC, destiné au monde du jeu et disposant des pilotes adaptés. Cette API est obsolète depuis la version 8.

- **Xinput** : gère les entrées manette Xbox 360. API de nouvelle génération qui remplace DirectInput. Supporté par les contrôleurs compatibles avec la Xbox 360.

- **DirectPlay** : facilite la communication sur un réseau local ou non. N'est plus mis à jour depuis la version 8.
- **DirectSound** : permet de jouer et d'enregistrer des sons au format PCM.
- **DirectSound3D (DS3D)** : permet de jouer des sons positionné dans un espace 3D. Cette API est obsolète.
- **Xaudio** : API sonores remplaçant DirectSound. Tire parti de la vitesse des microprocesseurs modernes afin de mixer un très grand nombre de canaux.
- **XAct** : API sonore haut niveau, construite à partir de Xaudio, permet de faciliter la programmation du son, avec notamment son propre format de banque sonore.
- **DirectMusic** : pour jouer des morceaux produit à l'aide de l'outil DirectMusic Producer.
- **DirectX Media** : comprend DirectAnimation pour l'animation web 2D. N'est plus mis à jour.
- **DirectShow** : pour manipuler les contenus multimédia, audio et vidéo, streamé ou non. Depuis avril 2005, DirectShow ne fait plus partie du SDK DirectX et a été déplacé dans le Platform SDK.
- **DirectX Transform** pour des effets web, et Direct3D Retained Mode pour faire de la 3D avec une API de plus haut niveau. DirectShow contient aussi DirectX plugins pour effectuer des opérations sur les signaux audio et DirectX Video Acceleration (DXVA) pour bénéficier d'une décompression matérielle des contenus vidéo compressés.
- **DirectX Media Objects** : composants additionnels pour des applications en rapport avec les streaming, encodeurs, décodeurs, effets.
- **DirectSetup** : permet de faciliter le déploiement des composants DirectX. Les fonctionnalités proposées par DirectX sont exposées sous forme d'interfaces et objet COM. Par ailleurs, des objets managés ont aussi été créés au-dessus de certaines parties de DirectX, comme managed DirectX (s'appuyant sur Direct3D 9 et des portions de DirectSound) ainsi que la bibliothèque graphique de XNA1.
- **Direct Compute** : API apparue lors du lancement de DirectX 10, visant à utiliser la puissance de calcul des GPU de dernière Génération.

API pour les jeux web

Un jeu web est un jeu qui se joue par l'intermédiaire d'un navigateur et cela sans avoir besoin d'installer et / ou de télécharger quelconque logiciel. Le jeu se joue directement sur le site Internet.

Voici la liste des APIs et des technologies pour les jeux web :

- ***Full Screen API*** : Cette API simple permet à votre jeu de fonctionner en plein écran et d'offrir ainsi une expérience plus immersive.
- ***Gamepad API*** : Permet aux joueurs d'utiliser des manettes ou d'autres contrôleurs.
- ***Web Audio API*** : Cette API pour contrôler la lecture, la synthèse et la manipulation de l'audio à partir du code JavaScript vous permet de créer des effets sonores impressionnants, de jouer et de manipuler de la musique en temps réel.

- ***IndexedDB*** : Une API puissante, permettant de maintenir les données de l'utilsateur stockées sur son ordinateur ou son appareil. Une bonne façon de sauvegarder l'état d'un jeu ou autre information localement de telle sorte qu'elle ne sera pas téléchargée à chaque fois qu'on aura besoin d'elle. Aussi utile pour rendre le jeu utilisable même quand le joueur n'est pas connecté à internet (comme durant un long vol en avion...).

- ***Pointer Lock API*** : vous permet de bloquer la souris ou tout autre appareil de pointage à l'intérieur de l'interface de votre jeu, de telle sorte qu'au lieu de recevoir une position absolue du pointeur vous recevrez le delta des coordonnées. Ce qui vous donne une mesure plus précise de ce que fait l'utilisateur, mais aussi l'empêche d'envoyer accidentellement ses entrées ailleurs pour ne pas manquer des actions importantes.

- ***WebGL*** : Vous permet de créer avec haute performance, et l'accélération matériel de graphisme 3D (et 2D) à partir de contenu internet . C'est une implémentation qui permet la prise en charge web de OpenGL ES 2.0.

- ***WebRTC*** : L'API WebRTC (Real-Time Communications) vous donne le pouvoir de contrôler les données audio et vidéo, y compris la téléconférence, et de transmettre des données d'applications aux utilisateurs et entre utilisateurs.

- ***WebSockets*** : L'API WebSocket vous permet de connecter votre application ou site à un serveur pour transmettre des données en temps réel. Parfait pour les jeux d'action multi-joueurs, services de «chat» (discussion) et autres.

La combinaison de **XMLHttpRequest** et de l'API **File** vous permet d'envoyer et de recevoir toutes les sortes de données que vous voulez depuis un serveur Web. Il s'agit d'une bonne façon de réaliser diverses choses comme télécharger de nouveaux niveaux du jeu et les éléments de graphismes ou encore transmettre (pas en temps réèl) les informations de l'état du jeu vers votre serveur.

L'image ci-dessous montre l'identité visuelle des API cités plus haut



image 01 *: Identités visuelles de quelques API pour le jeu vidéo*

III.1.4 Les environnements de développement intégré (IDE)

Un environnement de développement est un ensemble d'outils pour augmenter la productivité des programmeurs qui développent des logiciels. Il comporte un éditeur de texte destiné à la programmation, des fonctions qui permettent, par pression sur un bouton, de démarrer le compilateur ou l'éditeur de liens ainsi qu'un débogueur en ligne, qui permet d'exécuter ligne par ligne le programme en cours de construction. Certains environnements sont dédiés à un langage de programmation en particulier.

L'objectif d'un environnement de développement est d'augmenter la productivité des programmeurs en automatisant une partie des activités et en simplifiant les opérations. Les environnements de développement visent également à améliorer la qualité de la documentation en rapport avec le logiciel en construction. Certains environnements de développement offrent également la possibilité de créer des prototypes, de planifier les travaux et de gérer des projets.

Un EDI comporte au moins :

- Une interface graphique ;
- Elle permet notamment de sélectionner les fichiers, définir les options, lancer les opérations ;
- Un éditeur de code source avec coloration syntaxique ;
- La configuration d'un compilateur ;
- D'un éditeur de liens ;
- Un constructeur (outil make intégré). Il passe les commandes au compilateur et à l'éditeur de lien avec les fichiers sources ou les modules objets en paramètres ;
- Un outil de déboguage ;

En option on peut aussi avoir

- Le support de plusieurs langages.
- Un détecteur d'erreurs de syntaxe.
- L'auto-complétion de code.
- Des plugins.
- Un outil de construction visuelle d'interfaces avec copier et coller de widgets.
- La navigation dans les classes.

Les EDI peuvent être dédiés à un langage de programmation ou être multi-langages. Dans le second cas l'éditeur adapte la coloration syntaxique au langage, en fonction de l'extension des fichiers ou du choix de l'utilisateur.

Pour automatiser encore la production d'application, d'autres outils peuvent être incorporés à un EDI :

- Panel d'arborescence des classes et leur membres : Elle permet d'accéder facilement aux éléments d'un programme.
- Concepteur visuel. En choisissant des composants graphiques dans des barres, on crée une interface par glisser-déposer.
- Aide contextuelle. Fournit la fonction et le mode d'utilisation de chaque élément que l'on désigne.
- Gestionnaire de version. Maintient les versions successives du code source.
- Profileur. Aide à optimiser le code et améliorer ses performances.

Les éditeurs de texte modernes comme Bracket, Visual Studio Code, disposent outre la coloration syntaxique, de fonctionnalités étendues comme l'auto-complétion, la correction de syntaxe. Cela s'impose donc aussi dans une EDI.

Voici une liste des EDIs gratuits et très populaires

Visual Studio Community

C++, C#, Basic, HTML 5. Devenu gratuit en 2014. Il est disponible pour Windows.

Eclipse.

C'est un logiciel en Java, une plateforme de développement multi-langages intégrant les mêmes outils. Vous pouvez ajouter vos propres outils.

C'est un logiciel difficile à appréhender et peu apprécié. On se plaint notamment des plugins instables, la lenteur, et la difficulté à accéder aux composants. Ses avantages sont une bonne gestion de SVN et CVS, une large gamme de langages supportés.

KDevelop.

Un EDI dédié principalement aux applications en C++ et qui supporte d'autres bibliothèques dont OpenCL. Pour Linux et Windows.

IntelliJ IDEA.

Il se distingue par son fond gris par défaut, réalise des applications Java et HTML 5 pour le bureau ou pour Android en version CE, et d'autres langages pour la version pro. Bon debogueur, support pour le refactoring. Pour Windows/Mac/Linux.

Aptana Studio.

Pour réaliser des Applications Web avec HTML 5, PHP, Ruby. Automatisation de la mise en ligne et support de Git. Il est disponible pour Windows.

Light Table.

Un éditeur évolué (basé sur CodeMirror et Node) qui intègre un moteur de rendu HTML (NWJS ex-Node-Webkit) et une sorte de serveur interne. Il est prévu pour les langages dynamiques comme ClojureScript, JavaScript, Python et peut être complété de plugins pour d'autres langages. Il permet l'évaluation de code et la connexion à un utilisateur virtuel pour tester le code et le modifier en cours d'exécution. L'intégration du navigateur permet de voir le résultat des modifications sans recharger la page ce qui est idéal pour un projet Node.js. Il est disponiblr pour Windows, Linux et Mac.

Net Beans.

Applications Java, PHP, C++. Dispose d'un correcteur de syntaxe, l'accès aux fichiers source n'apparait qu'au besoin et libère l'écran autrement. Il convient bien pour les applications Web. Il est disponible pour Windows/Mac/Linux.

XCode.

Fourni avec toute une suite logicielle (graphiques, audio, etc.) pour développeurs et programmeurs, il permet de créer des logiciels utilisant toutes les fonctionnalités de macOS et d'UNIX. Cet environnement peut être obtenu gratuitement sur le Mac App Store.

Permet de programmer avec les langages Objective-C, Ruby et Swift.

L’image ci-dessous montre l’identité visuelle des IDE citées plus haut



image 02 *: Logo de quelques IDE pour le développement de jeu vidéo*

III.2 Les moteurs de jeu

Un moteur de jeu est un ensemble de composants logiciels qui effectuent des calculs de géométrie et de physique utilisés dans les jeux vidéo. L’ensemble forme un simulateur en temps réel souple qui reproduit les caractéristiques des mondes imaginaires dans lesquels se déroulent les jeux. Le but visé par un moteur de jeu est de permettre à une équipe de développement de se concentrer sur le contenu et le déroulement du jeu plutôt que la résolution de problèmes informatiques.

Le moteur 3D crée des images par des calculs de projection, tandis que le moteur 2D construit l’image du jeu par empilement d’images matricielles. Le moteur son effectue le mixage des bruits et de la musique tout au long du jeu. Les possibilités de scriptage des moteurs de jeu permettent de simuler le comportement des personnages non-jouables avec peu ou pas de programmation et le moteur physique sert à appliquer des règles de physique telles que l’inertie ou la pesanteur dans le but d’obtenir des mouvements plus réalistes.

III.2.1 Moteur de jeu pour débutant

Stencyl

Stencyl est un outil de création de jeux 2D pour Linux, OS X et Windows, y compris pour les plateformes mobiles. Il permet de créer des jeux simples en glissant et déposant de différents blocs, chacun avec une fonction associée. Ainsi, pour créer un jeu avec Stencyl, on n'a pas besoin de connaissances en programmation. Les utilisateurs plus expérimentés peuvent tout de même écrire leur propre code pour modifier les options proposées par l'outil.

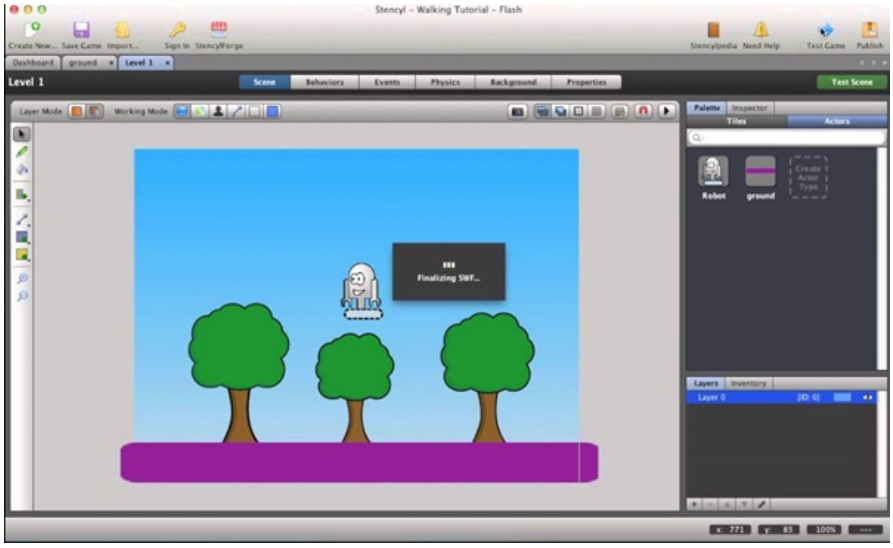


image 04 *: interface de Stencyl (source : https://fr.yeeply.com/)*

GDevelop

Disponible en open source, GDevelop peut être utilisé pour le développement de jeux 2D et 3D qui fonctionnent sous Windows, Linux et sur le web. Comme Stencyl, GDevelop dispose d'une interface glisser-déposer très facile d'utilisation, mais ce qui donne à cet outil un certain avantage est la possibilité de placer des événements sans avoir à écrire une seule ligne de code. Les évènements » permettent de répéter les conditions et actions pour les objets indiqués.



image 05 *: interface de GDevelop (source : https://fr.yeeply.com/)*

PyGame

Il s'agit d'une bibliothèque multimédia assez sympathique qui peut servir pour créer un jeu ou un autre programme. Afin de pouvoir en profiter pleinement, il faut avoir plus de connaissances en programmation qu'avec Stencyl et GDevelop, mais quelques bases en Python suffissent pour commencer.



image 06 *: Logo de PyGame (source : https://fr.yeeply.com/)*

Construct 2

Construct 2 est un programme qui vous laisse créer des jeux d'ordinateur HTML5 sans aucune expérience de programmation à l'aide d'un environnement de développement de type «drag and drop». La plupart des outils du programme peuvent être utilisés à partir de l'interface graphique sans écrire une seule ligne de code.

Construct 2 est conçu pour la création de jeux 2D et comprend beaucoup de ressources qui vous faciliteront la tâche, comme un moteur graphique qui fait en sorte que les items du jeu soient gouvernés par la loi de la gravité, ainsi que des éléments sonores comme des sprites, fonds d'écran et effets sonores. De plus, il est très simple d'ajouter des fichiers média externes.

La version gratuite vous laisse exporter vos créations vers le format HTML5, qui peut être joué sur n'importe quel navigateur peu importe la plateforme. La version pro payante, par contre, ajoute un outil d'exportation qui permet à vos jeux de s'exécuter sur Android ou iOS et de créer des fichiers exécutables qui peuvent être lancés sur un PC sans navigateur.

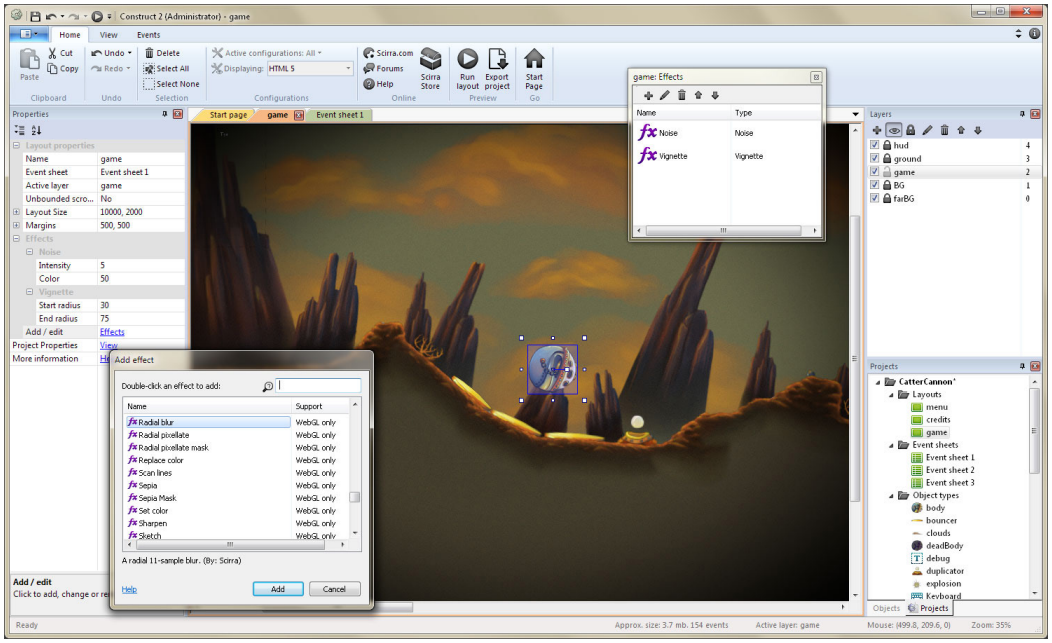


image 07 *: Interface de Construct 2 (source : https://www.scirra.com/)*

L'image ci-dessous montre l'identité visuelle des moteurs de jeu pour débutant cités plus haut :



1. Stencyl 2. GDevelop 3. Pygame 4. Construct 2

image 08 *: Identité visuelle des moteurs de jeu pour débutant cités plus haut*

III.2.2 Moteur de jeu pour les experts

Unity 3D

Considéré par de nombreux développeurs comme le meilleur outil de développement multiplateforme, Unity 3D permet de créer un jeu 3D ou 2D pour Xbox, PlayStation, Windows, Linux, iOS, Android, BlackBerry, la télévision connectée et, enfin, le web. Ce qui rend ce moteur de jeu si puissant est sans doute l'intégration du langage C#.



image 09 *: Interface de Unity 3D (source : https://fr.yeeply.com/)*

Unreal Engine

L'un des principaux concurrents d'Unity 3D, Unreal Engine utilise une conception orientée objet, ce qui en fait un outil extrêmement modulaire. Les développeurs peuvent l'utiliser gratuitement, mais à partir de 3000 dollars de chiffre d'affaires, ils doivent payer une redevance au studio Epic Games.



image 10 *: Interface de Unreal Engine (source : https://fr.yeeply.com/)*

CryEngine

C'est avec CryEngine qu'a été créée la célèbre **saga Crysis**, alors que la première version du logiciel a servi pour la conception de Far Cry. CryEngine est particulièrement adapté aux jeux de tir à la première personne.

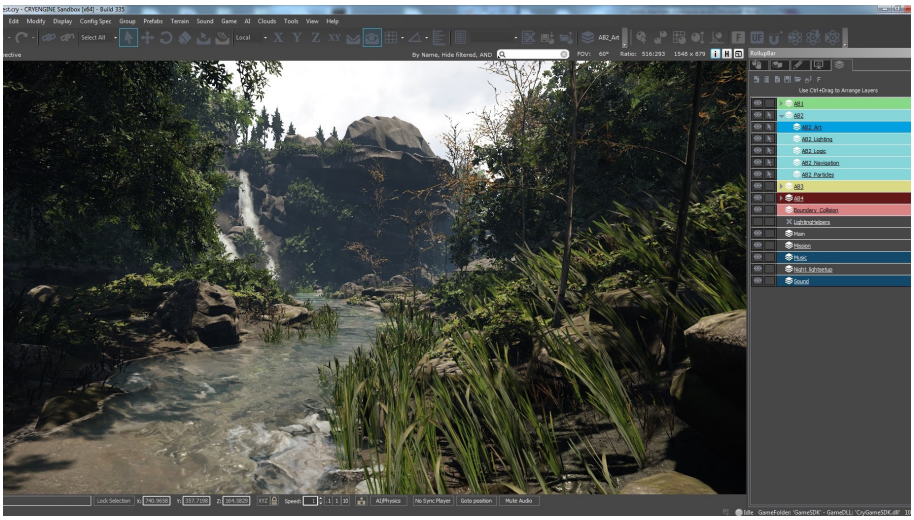


***image 11** : Interface de* CryEngine *(source : https://www.pcgamer.com/)*

GameMaker

GameMaker est un logiciel de développement de jeux vidéo créé par Mark Overmars à l'aide du langage de programmation Delphi. Mark Overmars utilise ce logiciel pour enseigner la programmation de Jeu vidéo à l'Université de Utrecht (Pays-Bas). La première version de Game Maker est sortie le 15 novembre 1999.

Game Maker a été conçu pour permettre aux développeurs amateurs de créer leurs propres jeux vidéo sans devoir apprendre un langage de programmation avancé comme C++ ou Java. On le voit parfois désigné par ses initiales GM. L'interface est en anglais, mais simple d'utilisation, et utilise abondamment le drag and drop.

Les possibilités offertes par cette application sont très larges. Tous les genres de jeux (plate-forme, action, réflexion, RPG, aventure...) peuvent être réalisés. Il est également possible d'utiliser des graphismes 2D ou bien 3D. Le jeu créé peut même être multijoueur en ligne. Il est aussi possible de créer des utilitaires avec Game Maker.

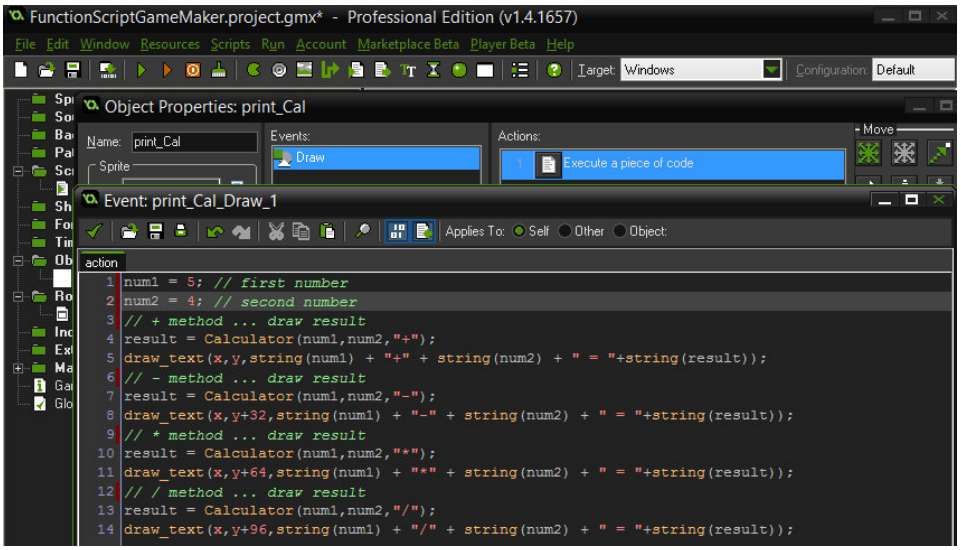


***image 12** : Interface de* GameMaker *(source : http://gamemakerlab.com/)*

Godot

Godot est un moteur de jeu multiplateforme capable de rendus 2D et 3D. Il est depuis janvier 2014 ouvert au public et disponible sous licence MIT, ce qui fait de lui un logiciel libre.

En tant qu'outil de création de jeux vidéo, Godot couvre de nombreux aspects de leur réalisation. L'utilisateur de Godot a la possibilité d'importer du contenu (images, son, modèles 3D), de créer des animations, de programmer le comportement de différents éléments du jeu grâce à un langage de script, de soumettre des objets à une physique (forces, collisions), de placer des sources de lumière, de créer un niveau constitué de tuiles au sein de l'éditeur, etc.

Il est disponible gratuitement aussi sous Windows, Linux et macOS. Il s'adresse principalement aux utilisateurs confirmés qui cherchent une solution solide pour développer des jeux indépendants. En cause, le langage de programmation utilisé est le GDScript, une variante du Python.



***image 13** : Interface de* Godot *(source : http://www.clubic.com/)*

Voici l'identité visuelle des moteurs de jeu pour expert cités plus haut

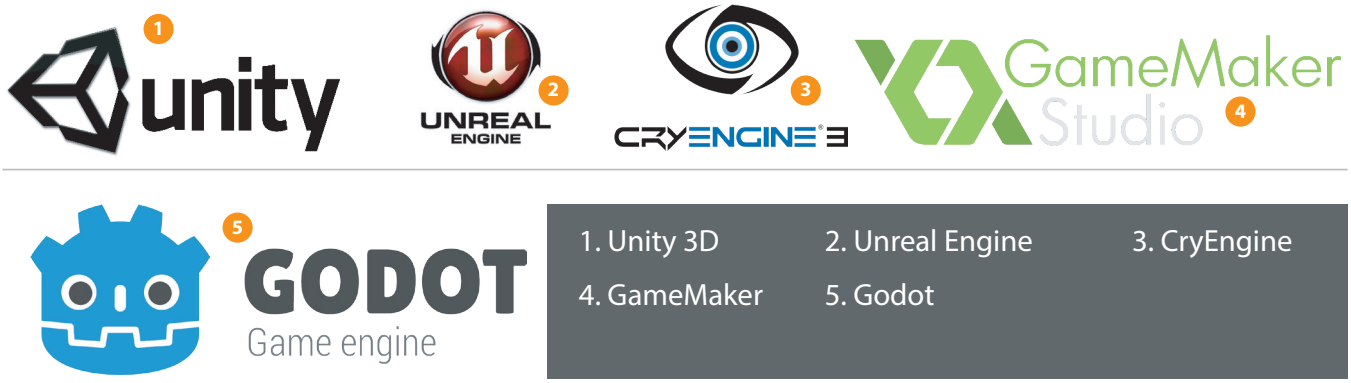


***image 14** : Identité visuelle des moteurs de jeu expert cités plus haut*

III.3 Les outils d'infographie

L'infographie en jeu vidéo couramment appelé **Game Art** permet de concevoir des ressources graphiques pour le jeu vidéo , que ce soit des environnements, des objets, des personnages ou des créatures.

L'infographiste en jeu vidéo (ou **game Artist**) réalise les décors et les personnages. il dessine, modélise, texture et anime. Son rôle est d'effectuer tous les graphismes et animations qui seront utilisés au sein du jeu. Un infographiste en jeu vidéo peut être amené à réaliser différents métiers au cours de sa carrière en fonction de sa spécialisation, il peut aussi bien rester infographiste « généraliste » au sein d'un petit studio, ou bien devenir Concept artist, character designer, environment designer, infographiste 3D, infographiste 2D, game artist modeleur, sculpteur digital, voir même Directeur artistique.

III.3.1 Logiciels 2D

Dans le domaine de l'infographie, un logiciel 2D (ou logiciels d'édition d'image, logiciel de graphisme ou logiciel de retouche d'image) est un programme informatique ou une collection de programmes permettant de manipuler et de traiter des formes, dessins, vecteurs ou images sur un ordinateur, éventuellement via une tablette graphique mais aussi un écran tactile comme le logiciel **ardoise**.

Ces logiciels, plus ou moins spécialisés sont utilisés par les artistes plasticiens, ou pour la retouche photo ou encore par des designers, architectes ou ingénieurs pour modéliser et présenter leurs projets, ou dans le cinéma et partout où l'image peut avoir une importance.

Un nombre croissant de ces logiciels sont disponibles en open-source (GIMP et Inkscape) sont les plus complets.

Adobe Illustrator

Adobe Illustrator est un logiciel de création graphique vectorielle. Il fait partie de la gamme Adobe et peut être utilisé indépendamment ou en complément de Photoshop, il offre des outils de dessin vectoriel puissants. Les images vectorielles sont constituées de courbes générées par des formules mathématiques. L'un des outils principaux d'Illustrator étant « la plume » qui permet de tracer des courbes à l'aspect parfait grâce au placement de points d'ancrage et de tangentes qui vont en modifier la courbure. Un des avantages des images vectorielles est qu'elles sont indépendantes de la résolution, c'est-à-dire qu'elles ne perdent pas en qualité lorsqu'on les agrandit. Adapté aussi bien à la création de document papier qu'à celle d'illustrations pour Internet (logos, affiches, etc.) ce logiciel est orienté vers le marché professionnel, il intègre de nombreuses options propres à améliorer la productivité.

FreeHand

FreeHand est un logiciel de création graphique vectorielle 2D de Adobe (anciennement Macromedia FreeHand).

FreeHand constitue l'outil idéal pour les dessinateurs qui créent et produisent des illustrations et des mises en page sur papier ou sur le Web. Il est parfaitement adapté à la création de storyboards, de clips Macromedia Flash MX, ou encore à la production de documents multipages et l'édition.

CorelDraw

CorelDRAW est une suite graphique développée par l'éditeur de logiciels Corel depuis 1989. À l'origine, il s'agissait du logiciel de dessin vectoriel Corel DRAW. Au fil du temps, d'autres logiciels tels que Corel PhotoPaint et Corel R.A.V.E ont été ajoutés et le logiciel est devenu une suite graphique.

CorelDRAW est le concurrent direct de Adobe Illustrator. De nombreux graphistes indépendants et agences de communication travaillant sur PC ont adopté le logiciel de Corel du fait de son prix attractif mais aussi parce qu'il se révèle parfois plus simple, rapide et efficace dans la manipulation des objets graphiques.

CorelDRAW est notamment utilisé par les créateurs de South Park afin de réaliser les illustrations 2D pour ensuite êtres animées en utilisant Maya.

Les fichiers produits par CorelDRAW (extension CDR) peuvent être importés par LibreOffice (version 3.6 et supérieure).

Photoshop

Édité par Adobe, il est principalement utilisé pour le traitement de photographies numériques, mais sert également à la création d'images.

Il travaille essentiellement sur images matricielles car les images sont constituées d'une grille de points appelés pixels. L'intérêt de ces images est de reproduire des gradations subtiles de couleurs.

Photoshop possède son propre format de fichier (extension psd). Celui-ci permet de conserver distincts les différents calques formant l'image afin de les manipuler séparément. Le programme accepte également d'importer et d'exporter des fichiers d'image dans les formats les plus courants (extensions : gif, jpg, tif, png, etc.).

Inkscape

Inkscape est un logiciel libre de dessin vectoriel sous licence GNU GPL. Il gère des fichiers conformes avec les standards XML, SVG et CSS du W3C.

Il s'agit d'un logiciel multiplate-forme, disponible notamment sous Windows, Mac OS X (sous X11), GNU/Linux et FreeBSD.

Il a des fonctionnalités assez similaires aux logiciels propriétaires CorelDRAW et Adobe Illustrator.

DrawPlus

Serif DrawPlus constitue une expérience de dessin et de retouche enrichissante. Des outils de dessin vectoriel simples vous permettent de créer des lignes, des formes et des courbes personnalisées. Ils sont accompagnés d'outils avancés pour les dessins, les fusions et les dégradés ciblés. Et comme l'image finale est un dessin vectoriel, vous pouvez facilement redimensionner votre illustration sans perte de qualité.

Serif DrawPlus donne vie à vos idées créatives au moyen de formes complexes, de couleurs et d'effets. Grâce aux nouveaux outils, vous pouvez tracer des arcs et des triangles. Un nouvel onglet Pochoirs propose une large gamme de modèles prêts à l'emploi et parfaits pour ajouter un impact à toutes vos créations.

Voici l'identité visuelle des logiciels 2D cités plus haut

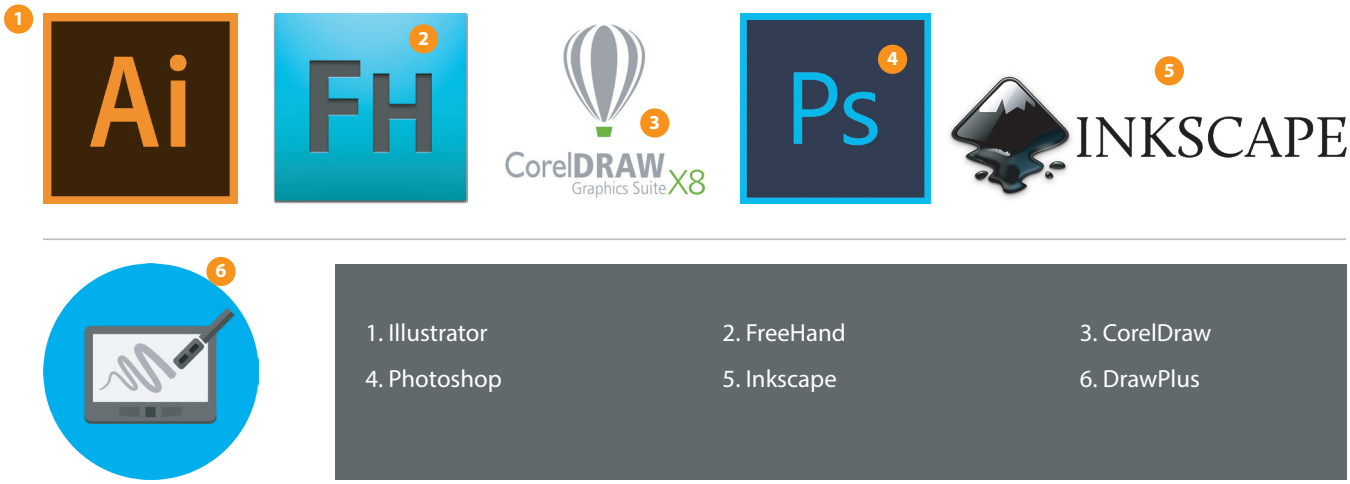


***image 15** : Identité visuelle des logiciels 2D cité plus haut*

III.3.2 Logiciels 3D

Un logiciel 3D ou logiciel de modélisation tridimensionnelle, ou modeleur 3D, est un logiciel qui sert à créer des scènes 3D, composées de formes complexes, ou objets, en trois dimensions à partir de primitives de bases ou de définition analytique. Les modeleurs 3D sont utilisés aussi bien dans l'industrie en conception assistée par ordinateur que par les infographistes qui réalisent des scènes dédiées aux jeux vidéo, à la réalisation d'animations pour le cinéma et à la création de présentations ou d'environnements de réalité virtuelle.

Les images réalisées en 3D s'illustrent dans bien des domaines : publicité, web, habillage télévisuel et cinématographique, effets spéciaux en tout genre, architecture, paysage, jeu video, video game art, video game design, infographie 3D, animation 3D…

Il existe une multitude de logiciels 3D. En voici quelques-uns :

Blender

Blender est un logiciel libre et gratuit de modélisation, d'animation et de rendu en 3D, créé en 1995. Il est actuellement développé par la Fondation Blender.

Il dispose de fonctions avancées de modélisation, de sculpture 3D, de dépliage UV, de texturage, de rigging, d'armaturage, d'animation 3D, et de rendu. Il gère aussi le montage vidéo non linéaire, la composition, la création nodale de matériaux, la création d'applications 3D interactives ou de jeux vidéo grâce à son moteur de jeu intégré (le Blender Game Engine), ainsi que diverses simulations physiques telles que les particules, les corps rigides, les corps souples et les fluides.

Disponible dans de nombreuses langues, Blender est également disponible sur plusieurs plates-formes telles que Microsoft Windows, Mac OS X, GNU/Linux, IRIX, Solaris, FreeBSD, SkyOS, MorphOS et Pocket PC. De plus, c'est un programme extensible (ajout de scripts) à l'aide du langage Python.

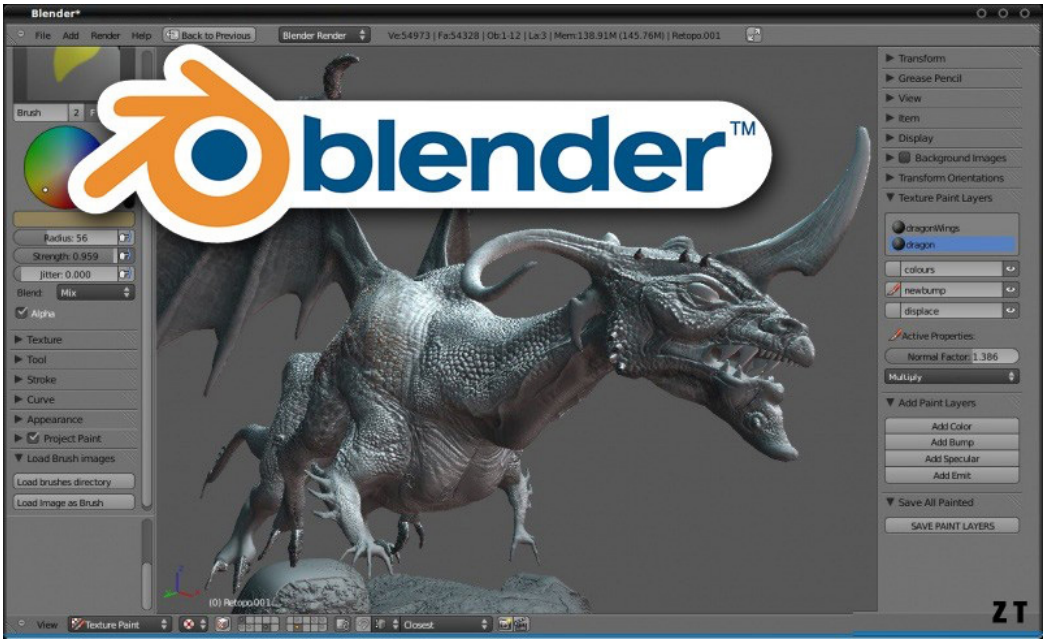


image 16 *: Interface de* Blender *(source : https://yggtorrent.com/)*

3ds Max

3ds Max (précédemment connu sous le nom «3D Studio MAX») est un programme de modélisation, d'animation, de compositing et de rendu 3D développé par la société Autodesk. Il est destiné aux concepteurs de jeux et de films, aux infographistes, aux spécialistes de l'animation et aux designers graphiques. Il est de ce fait le plus souvent utilisé dans les industries vidéo ludique, cinématographique et automobile. Il intègre de nombreuses fonctionnalités, le moteur de rendu Mental Ray par exemple, ou encore le moteur d'animation de personnage Character Studio.

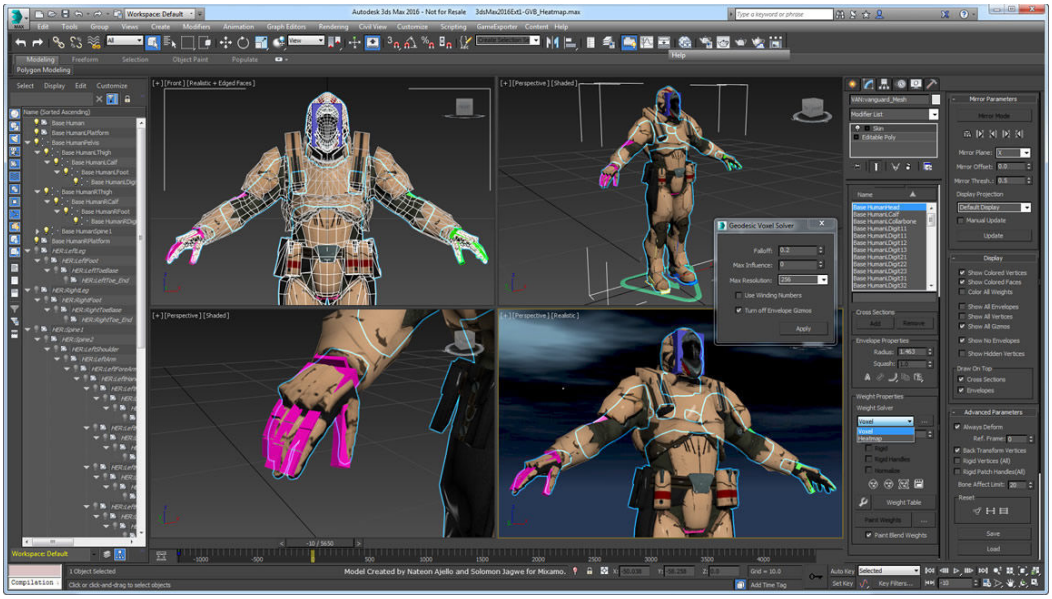


image 17 *: Interface de* 3ds Max *(source : http://www.directindustry.fr/)*

Cinema 4D

La personnalisation des animations devient de plus en plus facile avec Cinema4D, par exemple l'ajout de cheveux ou fourrure à des personnages, éclairage des champs, traitement de volume et des outils de réduction de polygone, gestion des ombres et transparences. Puisque ce logiciel a été conçu pour la réalisation de films, pour la télévision, la publicité, la science, l'architecture et d'autres domaines, Cinema4D fournit des résultats dignes d'un professionnel. Cela est valable pour l'emballage, l'architecture et la modélisation 3D. Le logiciel supporte les formats suivants : 3D Studio, Biovision BVH/BVA, DEM, DXF, Direct 3D, FBX, Lightwave, Monzoom, Quickdraw 3D, STL, Shockwave 3D, UZR, VMRL, Wavefront OBJ, TIFF, BodyPaint 3D, Targa TGA, BMP, PICT, IFF, JPEG, RLA, RPF, SGI, PNG, JP2, QuickTime, AVI, After Effects, Final Cut,



image 18 *: Interface de* Cinema 4D *(source : https://www.lynda.com/)*

MakeHuman

MakeHuman est un logiciel libre de modélisation 3D de corps humains. Les modèles générés sont destinés à être importés dans des logiciels de modélisations 3D plus généraux comme Blender ou 3D Studio Max.

MakeHuman dispose d'une interface simple pour générer un corps de type humain en se basant sur des critères généraux tels que le sexe, la musculature, le poids ou l'âge. Il permet de plus d'affiner chaque partie du corps séparément et de modifier la pose du modèle à l'aide d'un système d'os. Il est possible d'enregistrer les poses et les corps générés dans un format spécifique ou de les exporter au format Wavefront OBJ ou Collada.

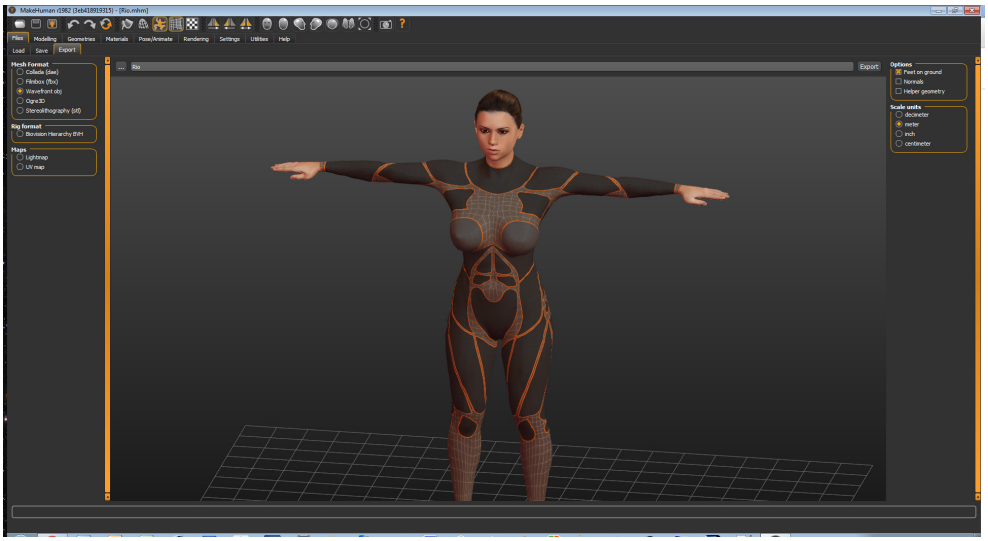


image 19 *: Interface de* MakeHuman *(source : https://forums.highfidelity.com/)*

Maya

Maya, qui veut dire « illusion » en sanskrit, est un logiciel commercial réputé pour les images de synthèse, développé par la société Alias Systems Corporation. Il est fortement utilisé par l'industrie du cinéma, principalement sous Linux.

Maya fonctionne aussi sous Mac OS X et Microsoft Windows. Le logiciel utilise intensivement le langage MEL, permettant à ses utilisateurs de personnaliser et d'ajouter des fonctionnalités sans avoir besoin de connaître le langage C ou C++. Il est toutefois possible de créer des extensions plus complexes grâce à une API utilisant le C++.

Maya intègre les technologies de rendu avancé, d'effets visuels, d'animation et de modélisation les plus sophistiqués dans une solution de travail complète. C'est le logiciel 3D le plus complet actuellement sur le marché, il est capable de gérer des données volumineuses et de produire des images de très haute qualité et s'intègre dans des pipelines de production complexes.



***image 20** : Interface de* Maya *(source : http://www.cgmeetup.net/)*

Wings 3D

Wings 3D est un logiciel libre de modélisation polygonale en trois dimensions par surface de subdivision. Il s'inspire des logiciels Nendo et Mirai de Izware. Wings 3D est disponible sous de nombreuses plate-formes, y compris GNU/Linux, Mac OS X et Windows. Il est écrit dans le langage de programmation Erlang et utilise une console virtuelle.

Il est conçu pour la modélisation et le texturage de modèles 3D, grâce à son interface relativement simple d'utilisation. À l'heure actuelle, le logiciel ne permet pas de faire des animations. Il dispose d'un moteur de rendu OpenGL et il est capable d'exporter les modèles dans de nombreux formats comme 3D Studio (3DS), Wavefront (OBJ), Nendo (NDO), VRML (WRL), Renderware (RWX), FBX (sur Windows et Mac OS X), Yafray et Toxic et il peut être directement importé par Blender. Il peut importer le format 3D Studio (3DS), Nendo (NDO), Wavefront (OBJ), FBX (sur Windows et Mac OS X) et Adobe Illustrator 8 (AI). De plus, son système de plugin rend possible l'extension des capacités du logiciel : il est par exemple possible d'utiliser un moteur de rendu par lancer de rayon externe tel que Yafray ou PovRay.

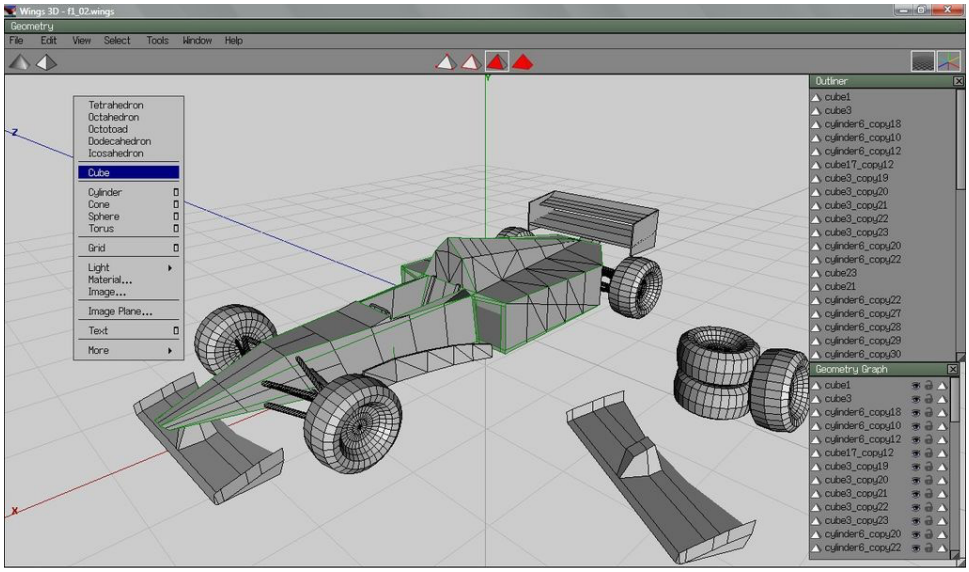
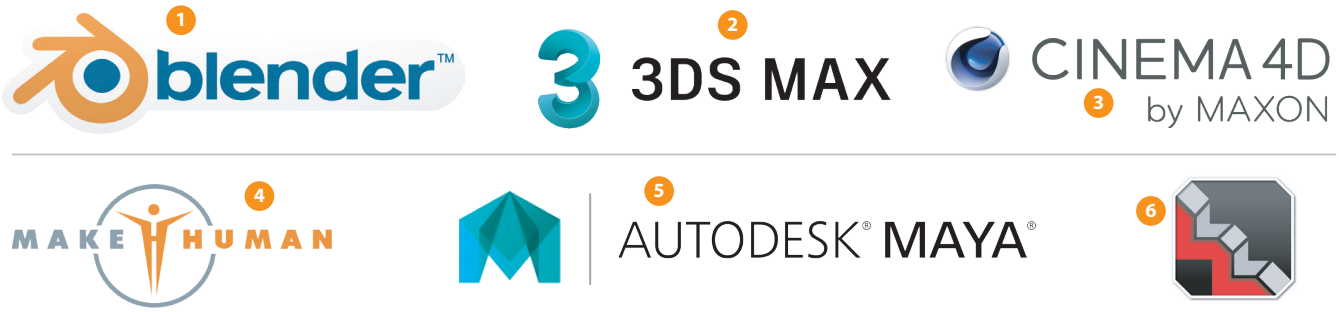


***image 21** : Interface de* Wings 3D *(source : https://www.topbestalternatives.com/)*

Voici l'identité visuelle des logiciels 3D cités plus haut



1. Blender
2. 3ds Max
3. Cinema 4D
4. MakeHuman
5. Maya
6. Wings 3D

***image 22** : Identité visuelle des logiciels 3D cités plus haut*

III.3.3 UML en jeu vidéo

Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.

UML est utilisé pour spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet. UML offre un standard de modélisation, pour représenter l'architecture logicielle. Les différents éléments représentables sont :

- Activité d'un objet/logiciel
- Acteurs
- Processus
- Schéma de base de données
- Composants logiciels
- Réutilisation de composants

Grâce aux outils de modélisation UML, il est également possible de générer automatiquement tout ou partie du code d'une application logicielle, par exemple en langage Java, à partir des divers documents réalisés.

Voici quelques logiciels de modélisation UML.

StarUML

Grâce à ce logiciel, il vous sera possible de concevoir des classes, des objets et des acteurs et d'y définir nombre d'attributs. Vous pourrez également créer et ajouter vos propres plugins, les langages de programmation Com, C++, C# et Delphi sont pris en charge. StarUML supporte également l'architecture MDA qui offre comme avantage la personnalisation des profils UML, les modèles de documents ainsi que le code MDA.

En somme, StarUML est complet, robuste et présente une pléthore d'outils et de paramètres. Il est toutefois dédié principalement aux utilisateurs chevronnés et aux projets complexes.

ArgoUML

ArgoUML est un logiciel libre de création de diagrammes UML. Programmé en Java, il est édité sous licence EPL 1.0. Il est multilingue, supporte la génération de code et l'ingénierie inverse.

ArgoUML propose aux développeurs un outil de représentation UML, leader de la scène open source. L'application est compatible avec tous les diagrammes UML 1.4 standards et fonctionne sur n'importe quelle plateforme Java. ArgoUML est livré avec des profils pour le développement d'application C++ et Java. Il supporte les diagrammes de classe, les diagrammes d'état, les diagrammes «Use Case» ou encore les diagrammes de séquence, et plus encore. Enfin, ArgoUML offre des fichiers de sauvegarde ouverts basés sur du XML, et peut exporter les différents diagrammes aux formats GIF, PNG, PostScript, Encapsulated PS, PGML et SVG.

Visual Paradigm

Visual Paradigm est, comme son nom le laisse supposer, un logiciel permettant aux programmeurs de mettre en place des diagrammes UML. Disposant d'un outil créant des rapports personnalisables aux formats PDF, Word ou HTML afin de les partager et les publier sur Internet, cette application est compatible avec de nombreuses applications, standards et environnements. Ainsi, vous pourrez générer notamment des diagrammes de séquences ou de cas d'utilisation et ainsi produire du code source dans de nombreux langages comme le Java ou encore le C++, ou bien faire l'inverse, générer des diagrammes à partir de code déjà existant.

BOUML

BOUML est un logiciel de création de diagrammes UML. Programmé en C++ et Qt. Auparavant distribué gratuitement sous licence GPL, puis commercialisé pour les versions 5.0 à 6.12, il est de nouveau distribué gratuitement depuis la version 7.0 (mais pas intégralement sous licence GPL). Il est multilingue, supporte la génération de code et la rétro-ingénierie.

- Les caractéristiques du logiciel sont :
- Permet de dessiner des diagrammes suivant la norme UML 2.0.
- Est capable d'effectuer de la rétro-ingénierie.
- Est multiplateforme.
- Peut être utilisé en français (depuis la version 4.15.1).

Voici l'identité visuelle des logiciels de modélisation UML cités plus haut



***image 23** : Identité visuelle des logiciels de modélisation UML cités plus haut*

Sources

https://www.all-gamers.fr/les-outils-de-creation-de-jeux-video/

https://fr.wikipedia.org/wiki/Programmation_de_jeux_vidéo

https://jeux.developpez.com/tutoriels/jeux-video/

https://openclassrooms.com/courses/un-jeu-en-ligne-la-marche-a-suivre

https://developer.mozilla.org/fr/docs/Jeux/Introduction

https://fr.wikipedia.org/wiki/Développeur_de_jeux_vidéo#Langage_de_programmation

https://fr.wikipedia.org/wiki/Interface_de_programmation

https://www.scriptol.fr/programmation/edi.php

https://fr.wikipedia.org/wiki/Liste_de_moteurs_de_jeu

https://fr.yeeply.com/blog/environnements-developpement-creer-jeu-video/

https://www.yoyogames.com/gamemaker

http://www.clubic.com/telecharger-fiche433879-godot-engine-1.html

https://construct-2.fr.uptodown.com/windows

https://www.scirra.com/construct2

http://www.creajeux.fr/nos-formations-jeu-video/infographiste/

https://www.video2brain.com/fr/3ds-max

http://www.commentcamarche.com/download/telecharger-34083617-cinema4d

https://yggtorrent.com/torrent/application/formation/21899-formation+blender+2+7

http://www.directindustry.fr/prod/autodesk/product-14521-252473.html

https://www.lynda.com/CINEMA-4D-tutorials/CINEMA-4D-R15-New-Features/146013-2.html

https://forums.highfidelity.com/t/avatar-licence-marketplace/5316/58

http://www.cgmeetup.net/home/autodesk-maya-tips-and-tricks/

https://www.topbestalternatives.com/wings-3d/

https://fr.wikipedia.org/wiki/UML_(informatique)

http://www.01net.com/telecharger/windows/Multimedia/creation_graphique/fiches/204.html

https://fr.wikipedia.org/wiki/Macromedia_FreeHand

https://www.pcgamer.com/cryengine-steam/

http://gamemakerlab.com/gamemaker/how-to-write-script-function-in-gamemaker-tutorials