

**Les cartes graphiques**

Par

**David Lafrenière**

**Samuel Forcier**

Travail présenté à

Louis Marchand

Jonathan Gareau

Dans le cadre du cours

420-DC1-DM : Documentation informatique

et gestion de documents

Cégep de Drummondville

xx xxxxxxx 2013

# Table des matières

[Table des matières 2](#_Toc353542848)

[Introduction 4](#_Toc353542849)

[Les technologies propriétaires NVIDIA 5](#_Toc353542850)

[3D Vision surround 5](#_Toc353542851)

[Qu’est ce que la technologie 3D Vision: 5](#_Toc353542852)

[Qu’elle est le materiel necessaire au fonctionnement : 5](#_Toc353542853)

[Compatibilité des jeux : 6](#_Toc353542854)

[SLI 6](#_Toc353542855)

[Qu’est ce que le SLI : 6](#_Toc353542856)

[Qu’elle est le materiel necessaire au fonctionnement : 6](#_Toc353542857)

[Historique : 7](#_Toc353542858)

[Modes d’utilisation de la technologie SLI : 8](#_Toc353542859)

[SFR 8](#_Toc353542860)

[AFR 8](#_Toc353542861)

[AFR de SFR 9](#_Toc353542862)

[SLI AA 9](#_Toc353542863)

[A quel point les performances sont améliorées par la technologie SLI? : 10](#_Toc353542864)

[CUDA(Compute Unified Device Architechture) 10](#_Toc353542865)

[PhysX 11](#_Toc353542866)

[[edit]Features 12](#_Toc353542867)

[[edit]APEX PhysX 12](#_Toc353542868)

[[edit]Hardware acceleration 13](#_Toc353542869)

[[edit]PPU 13](#_Toc353542870)

[[edit]GPU 13](#_Toc353542871)

[[edit]Real World Technologies analysis 14](#_Toc353542872)

[[edit]Use 15](#_Toc353542873)

[[edit]PhysX in video games 15](#_Toc353542874)

[[edit]PhysX in other software 16](#_Toc353542875)

[Comparatif 16](#_Toc353542876)

[Conclusion 17](#_Toc353542877)

[Réferences 18](#_Toc353542878)

# Table des illustrations

[Figure 1 : Commencement du SLI 8](#_Toc353543754)

[Figure 2 : Mode SFR du SLI 9](#_Toc353543755)

[Figure 3 :Mode AFR du SLI 10](#_Toc353543756)

[Figure 4 :Mode AFR de SFR du SLI 10](#_Toc353543757)

[Figure 5 :Anticrénelage du SLI 11](#_Toc353543758)

[Figure 6 :Comparatif des gains SLI 11](#_Toc353543759)

# Introduction

# Les technologies propriétaires NVIDIA

## 3D Vision surround

### Qu’est ce que la technologie 3D Vision:

3D Vision de NVIDIA est une technologie propriétaire, elle peut se comparer a Eyefinity de AMD. 3D Vision sert à étendre les jeux sur plusieurs ecran, jusqu'à trois écrans a la fois, soit une résolution de 5760x1080. Cette technologie sert aussi a implanter le 3D.

### Qu’elle est le materiel necessaire au fonctionnement :

Premierement, il faut avoir un système d’exploitation parmis les suivant :

-Windows XP

-Windows Vista

-Windows 7

-Linux

Il faut aussi possedé une carte mere SLI ou bien avoir une carte NVIDIA ayant un double processeur graphique comme la GTX 590 ou la GTX 690. L’utilisation de trois écrans requiert aussi d’avoir trois connecteurs numériques (DVI, HDMI) ou trois connecteurs analogiques (VGA) et biensur trois écrans identiques. Pour afficher un image 3D, nous devons avoir des écran LCD de 120 hz compatible avec 3D Vision ou des projecteurs encore une fois compatible avec la technologie 3D d’NVIDIA, les projecteurs doivent etre du meme modèle et du meme constructeur.

Toutefois, si nous voulons jouer en 3D, il faut ajouter les lunettes 3D de NVIDIA.

### Compatibilité des jeux :

Plus de 600 jeux sont présentement compatible avec 3D Vision, si vous voulez voir la liste il suffie d’aller dans le panneau de configuration NVIDIA et de regarder la compatibilité des jeux. NVIDIA ne cesse de tester la compatibilité des jeux avec leur technologie 3D, ils sont noté selon leur compatibilité. Les notes attribués sont soit excellent, bon, correct ou non recommandé. Si le un jeux ne figure pas dans la liste de test de NVIDIA, il est quand meme possible de testé la compatibilité par soit meme.

## SLI

### Qu’est ce que le SLI :

La technologie SLI de NVIDIA est une plateforme qui permet de d’augmenter la performances graphique d’un ordinateur en combinant plusieurs carte graphiques NVIDIA.

### Qu’elle est le materiel necessaire au fonctionnement :

Pour commencer il faut avoir une carte mere certifié SLI et il faut aussi avoir soit deux ou trois cartes graphiques certifié SLI

### Historique :

La technologie SLI a été inventé en 1998 par 3DFX et leurs carte Voodoo 2. Au commencement, SLI voulais dire «Scan Line Interleaving» et fesais travailler chacun des processeurs graphiques sur un groupe de ligne, par exemple un processeur travaillais sur les lignes impaires et l’autre sur les lignes paires

Processeur 2

Processeur 1

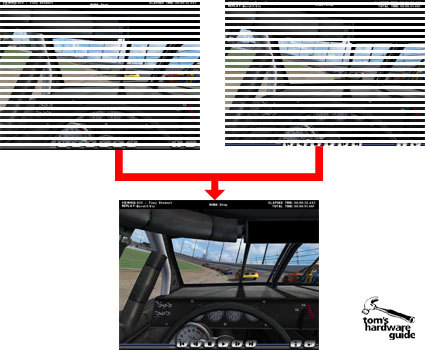


Figure 1 : Commencement du SLI

Lien http://www.tomshardware.com/reviews/nvidia-takes-a-walk-sli-side-double-graphics-processing,834-2.html

En 2001, NVIDIA achete 3dfx et sort un concept similaire et amélioré en 2004, le concept du SLI actuel (Scalable Link Interface).

### 

### **Modes d’utilisation de la technologie SLI :**

#### **SFR**

La premiere utilisation se nomme SFR pour Split Frame Rendering, ce mode fonctionne en séparant chacune des image afficher a l’écran en deux, une moitier est traité par le premier processeur graphique et l’autre moitier par le deuxieme.

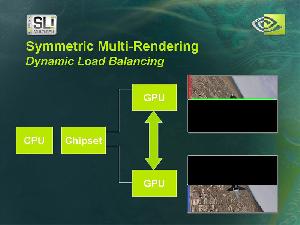


Figure 2 : Mode SFR du SLI

#### AFR

La deuxieme utilisation est appelé AFR pour Alternate Frame Rendering. Sont fonctionnement est simple, chaque carte graphique traite une image à la fois. Donc, pendant que processeur GPU#1 traite l’image 1, le GPU#2 commence déjà a traiter l’image 2, en resumé lorsque l’image 1 est afficher a l’écran la 2ieme image est déjà prete a etre afficher

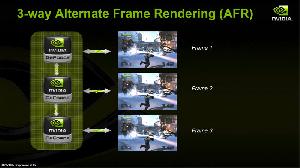


Figure 3 :Mode AFR du SLI

#### AFR de SFR

La troisieme utilisation est une combine des deux premiere. Celle-ci est utiliser en mode quad-SLI. Dans ce mode les deux premier GPU traite la premiere image comme en AFR séparement comme en SFR et pendant ce temps les deux autre GPU eux traite la 2ieme image comme en AFR séparement comme en SFR.



Figure 4 :Mode AFR de SFR du SLI

#### SLI AA

Le dernier mode d’utilisation est SLI AA pour anticrénelage et mieu connu sous le nom de «Anti-Aliasing », ce mode n’est pas fais pour amelioré la performances pour les jeux, mais il est fais pour amélioré la qualité d’image. Avec un seul GPU il est possible d’aller jusqu'à 8x anticrénelage, avec le SLI on peut aller jusqu'à 16x ou meme 32x dependement du mode SLI.

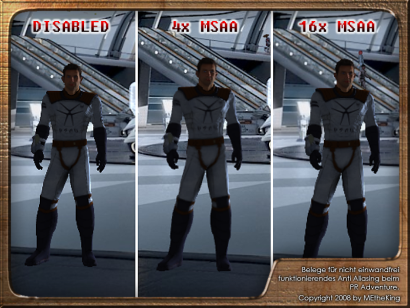


Figure 5 :Anticrénelage du SLI

Lien : <http://www.overclock.net/t/988582/what-is-aa-exactly>

### A quel point les performances sont améliorées par la technologie SLI? :

Les performances sont grandement amélioré, elle peuvent aller de 25% à 100% avec deux cartes graphiques et meme 200% d’amelioration avec trois carte graphiques.

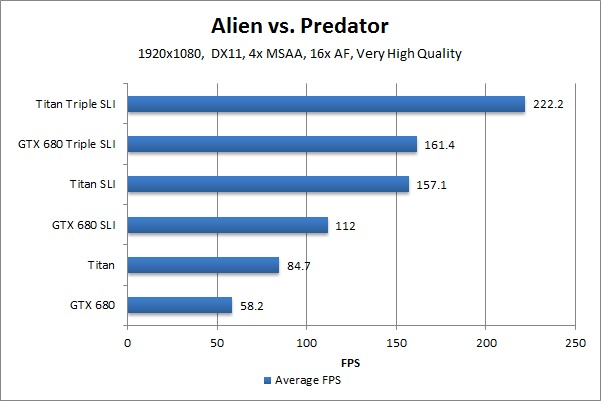


Figure 6 :Comparatif des gains SLI

## CUDA(Compute Unified Device Architechture)

La technologie CUDA est une architechture développer par NVIDIA, celle-ci permet de d’augmenter les performances de calcul d’un système en utilisant les processeurs graphiques. Les GPU peuvent en fait effectuer des calculs qui était fait auparant par le processeur. Plusieurs API (application programming interface) comme OpenCl, DirectX et Fortan utilisent la technologie CUDA. OpenCl pour Open Computing Language est un API de calcul de bas niveau qui utilise l’architechture CUDA. DirectX Compute de Microsoft utilise aussi CUDA et est supporté sur les GPU de la catégorie directX 10 et 11. Fortran est aussi utilisateur de CUDA.

CUDA n’est pas seulement utilisé pour les jeux. On l’utilise aussi pour faire des simulations de flux sanguin a l’université Harvard afin d’éviter des opérationspour investiger. On l’utilise aussi pour la simulation de molécule est pour le control de l’espace aérien. Avec CUDA, on peut maintenant envoyer des insctruction en C ou C++ directement au GPU, donc le langage assembleur n’est plus necessaire.

JE NE SAIS PAS TROP QUOI DIRE

## PhysX

PhysX est un moteuvr physique proprietaire à NVIDIA. Le nom PhysX peut aussi referer a physics processing unit expansion card. Il est principalement utiliser pour la physique dans les jeux video.

Video games supporting [hardware acceleration](http://en.wikipedia.org/wiki/Hardware_acceleration) by PhysX can be accelerated by either a PhysX PPU or a [CUDA](http://en.wikipedia.org/wiki/CUDA)-enabled [GeForce](http://en.wikipedia.org/wiki/GeForce) [GPU](http://en.wikipedia.org/wiki/Graphics_processing_unit) (if it has at least 256MB of dedicated VRAM), thus offloading physics calculations from the [CPU](http://en.wikipedia.org/wiki/Central_processing_unit), allowing it to perform other tasks instead. This typically results in a smoother gaming experience[*[citation needed](http://en.wikipedia.org/wiki/Wikipedia:Citation_needed" \o "Wikipedia:Citation needed)*] and additional visual effects.

What is known today as PhysX originated as a physics simulation engine called NovodeX. The multi-threaded engine was developed by Swiss company NovodeX AG. In 2004, [Ageia](http://en.wikipedia.org/wiki/Ageia" \o "Ageia) acquired NovodeX AG and began developing a hardware technology that could accelerate physics calculations, aiding the CPU. Ageia called the technology PhysX PPU (physics processing unit), and the SDK was renamed from NovodeX to PhysX.[[8]](http://en.wikipedia.org/wiki/PhysX#cite_note-8)

In 2008, Ageia was itself acquired by graphics technology manufacturer [Nvidia](http://en.wikipedia.org/wiki/Nvidia" \o "Nvidia). Nvidia started enabling PhysX hardware acceleration on its line of GeForce graphics cards[[9]](http://en.wikipedia.org/wiki/PhysX" \l "cite_note-9) and eventually dropped support for Ageia PPUs.[[10]](http://en.wikipedia.org/wiki/PhysX#cite_note-NoSupport-10)

## [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=2)]Features

PhysX is a multi-threaded physics simulation [SDK](http://en.wikipedia.org/wiki/Software_development_kit) available for [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows), [Mac OS X](http://en.wikipedia.org/wiki/Mac_OS_X), [Linux](http://en.wikipedia.org/wiki/Linux), [PlayStation 3](http://en.wikipedia.org/wiki/PlayStation_3), [Xbox 360](http://en.wikipedia.org/wiki/Xbox_360) and [Wii](http://en.wikipedia.org/wiki/Wii). It supports rigid body dynamics, soft body dynamics, [ragdolls](http://en.wikipedia.org/wiki/Ragdoll_physics) and character controllers, vehicle dynamics, volumetric fluid simulation and cloth simulation including tearing and pressurized cloth.

### [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=3)]APEX PhysX

[Nvidia](http://en.wikipedia.org/wiki/Nvidia) APEX technology is a multi-platform scalable dynamics framework first introduced in [*Mafia II*](http://en.wikipedia.org/wiki/Mafia_II) in August 2010.[[11]](http://en.wikipedia.org/wiki/PhysX#cite_note-APEX-11) Nvidia's APEX engine comprises the following features: APEX Destruction, APEX Clothing, APEX Particles, APEX Vegetation, and APEX Turbulence.[[12]](http://en.wikipedia.org/wiki/PhysX#cite_note-12)

## [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=4)]Hardware acceleration

[](http://en.wikipedia.org/wiki/File:Mafia_physx.jpg)

[http://bits.wikimedia.org/static-1.21wmf11/skins/common/images/magnify-clip.png](http://en.wikipedia.org/wiki/File:Mafia_physx.jpg)

The top screenshot shows how debris is simulated in [Mafia II](http://en.wikipedia.org/wiki/Mafia_II) when PhysX is turned to the highest level in the game settings. The bottom screenshot shows a similar scene with PhysX turned to the lowest level.

### [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=5)]PPU

A [physics processing unit](http://en.wikipedia.org/wiki/Physics_processing_unit) (PPU) is a processor specially designed to alleviate the calculation burden on the CPU, specifically calculations involving physics. PPU cards with PhysX support were available from the manufacturers [ASUS](http://en.wikipedia.org/wiki/ASUS), [BFG Technologies](http://en.wikipedia.org/wiki/BFG_Technologies)[[13]](http://en.wikipedia.org/wiki/PhysX#cite_note-13) and [ELSA Technology](http://en.wikipedia.org/wiki/ELSA_Technology). Beginning with version 2.8.3 of the PhysX SDK, support for PPU cards was dropped, and PPU cards are no longer manufactured.[[10]](http://en.wikipedia.org/wiki/PhysX#cite_note-NoSupport-10)

### [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=6)]GPU

After [Nvidia](http://en.wikipedia.org/wiki/Nvidia" \o "Nvidia)'s acquisition of [Ageia](http://en.wikipedia.org/wiki/Ageia" \o "Ageia), PhysX development turned away from PPU extension cards and focused instead on the [GPGPU](http://en.wikipedia.org/wiki/GPGPU) capabilities of modern [GPUs](http://en.wikipedia.org/wiki/Graphics_processing_unit). A graphics processing unit or GPU (also occasionally called visual processing unit or VPU) is a dedicated graphics rendering device for a personal computer, workstation or game console. Modern GPUs are very efficient at manipulating and displaying computer graphics, and their highly parallel structure makes them more effective than general-purpose CPUs for a range of complex algorithms, such as accelerating physical simulations using PhysX. A GPU can sit on top of a video card, or it can be integrated directly into the motherboard. More than 90% of new desktop and notebook computers have integrated GPUs.

Any [CUDA](http://en.wikipedia.org/wiki/CUDA)-ready GeForce graphics card (series 8 and newer, with a minimum of 256MB of video memory[[14]](http://en.wikipedia.org/wiki/PhysX" \l "cite_note-nvidia.com-14)) can take advantage of PhysX without the need to install a dedicated PhysX card.

Versions 186 and newer of the ForceWare drivers disable PhysX hardware acceleration if a GPU from a different manufacturer, such as [AMD](http://en.wikipedia.org/wiki/Advanced_Micro_Devices), is present in the system.[[14]](http://en.wikipedia.org/wiki/PhysX#cite_note-nvidia.com-14) Representatives at Nvidia stated to customers that the decision was made due to development expenses, and for quality assurance and business reasons.[[15]](http://en.wikipedia.org/wiki/PhysX#cite_note-15) This decision has caused a backlash from the community that led to the creation of a community patch for Windows 7, circumventing the GPU check in Nvidia's updated drivers. To counter this patch, Nvidia implemented a time bomb in driver versions 196 and 197 that slowed down hardware accelerated PhysX and reversed the gravity,[[16]](http://en.wikipedia.org/wiki/PhysX#cite_note-16) but an updated version of the patch removed all unwanted effects.[[17]](http://en.wikipedia.org/wiki/PhysX#cite_note-17)

## [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=7)]Real World Technologies analysis

On 5 July 2010, Real World Technologies published an analysis[[18]](http://en.wikipedia.org/wiki/PhysX" \l "cite_note-18) of the PhysX architecture. According to this analysis, most of the code used in PhysX applications at the time was based on [x87](http://en.wikipedia.org/wiki/X87)instructions without any multi-threading optimization. This could cause significant performance drops when running PhysX code on the CPU. The article suggested that a PhysX rewrite using [SSE](http://en.wikipedia.org/wiki/Streaming_SIMD_Extensions) instructions may substantially lessen the performance discrepancy between CPU PhysX and GPU PhysX.

In response to the Real World Technologies analysis, Mike Skolones, product manager of PhysX, said[[19]](http://en.wikipedia.org/wiki/PhysX" \l "cite_note-19) that SSE support had been left behind because most games are developed for [consoles](http://en.wikipedia.org/wiki/Video_game_console) first and then ported to the PC. As a result, modern computers run these games faster and better than the consoles even with little or no optimization. Senior PR manager of Nvidia, Bryan Del Rizzo, explained that multi-threading had already been available with CPU PhysX 2.x and that it had been up to the developer to make use of it. He also stated that automatic multithreading and SSE would be introduced with version 3 of the PhysX SDK.[[20]](http://en.wikipedia.org/wiki/PhysX#cite_note-20)

PhysX SDK 3.0 was released in May 2011 and represented a significant rewrite of the SDK, bringing improvements such as more efficient multithreading and a unified code base for all supported platforms.[[2]](http://en.wikipedia.org/wiki/PhysX#cite_note-sdk3-2)

## [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=8)]Use

### [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=9)]PhysX in video games

*See also:*[*List of games with hardware-accelerated PhysX support*](http://en.wikipedia.org/wiki/List_of_games_with_hardware-accelerated_PhysX_support)

PhysX technology is used by [game engines](http://en.wikipedia.org/wiki/Game_engine) such as [Unreal Engine](http://en.wikipedia.org/wiki/Unreal_Engine) (version 3 onwards), [Unity 3D](http://en.wikipedia.org/wiki/Unity_(game_engine)), [Gamebryo](http://en.wikipedia.org/wiki/Gamebryo" \o "Gamebryo), [Vision](http://en.wikipedia.org/wiki/Vision_Engine) (version 6 onwards), Instinct Engine,[[21]](http://en.wikipedia.org/wiki/PhysX#cite_note-21) [Panda3D](http://en.wikipedia.org/wiki/Panda3D), [Diesel](http://en.wikipedia.org/wiki/Diesel_(game_engine)), [Torque](http://en.wikipedia.org/wiki/Torque_(game_engine)), [HeroEngine](http://en.wikipedia.org/wiki/HeroEngine" \o "HeroEngine) and[BigWorld](http://en.wikipedia.org/wiki/BigWorld).[[22]](http://en.wikipedia.org/wiki/PhysX#cite_note-22)

As one of the handful of major physics engines, it is used in many games, such as *[Bulletstorm](http://en.wikipedia.org/wiki/Bulletstorm" \o "Bulletstorm)*, [*Need for Speed: Shift*](http://en.wikipedia.org/wiki/Need_for_Speed:_Shift), *[Castlevania: Lords of Shadow](http://en.wikipedia.org/wiki/Castlevania:_Lords_of_Shadow" \o "Castlevania: Lords of Shadow)*, [*Mafia II*](http://en.wikipedia.org/wiki/Mafia_II), [*Alice: Madness Returns*](http://en.wikipedia.org/wiki/Alice:_Madness_Returns), [*Batman: Arkham City*](http://en.wikipedia.org/wiki/Batman:_Arkham_City)etc. Most of these games use the CPU to process the physics simulations.

Video games with optional support for hardware-accelerated PhysX often include additional effects such as tearable cloth, dynamic smoke or simulated particle debris.[[23]](http://en.wikipedia.org/wiki/PhysX#cite_note-23)[[24]](http://en.wikipedia.org/wiki/PhysX#cite_note-24)[[25]](http://en.wikipedia.org/wiki/PhysX#cite_note-25)

### [[edit](http://en.wikipedia.org/w/index.php?title=PhysX&action=edit&section=10)]PhysX in other software

Other software with PhysX support includes:

* [Active Worlds](http://en.wikipedia.org/wiki/Active_Worlds) (AW), a 3D virtual reality platform with its client running on Windows
* [Autodesk 3ds Max](http://en.wikipedia.org/wiki/Autodesk_3ds_Max), [Autodesk Maya](http://en.wikipedia.org/wiki/Autodesk_Maya) and [Autodesk Softimage](http://en.wikipedia.org/wiki/Autodesk_Softimage), [computer animation](http://en.wikipedia.org/wiki/Computer_animation) suites[[26]](http://en.wikipedia.org/wiki/PhysX#cite_note-26)[[27]](http://en.wikipedia.org/wiki/PhysX#cite_note-27)[[28]](http://en.wikipedia.org/wiki/PhysX#cite_note-28)
* [DX Studio](http://en.wikipedia.org/wiki/DX_Studio), an integrated development environment for creating interactive 3D graphics[[30]](http://en.wikipedia.org/wiki/PhysX#cite_note-30)
* [Futuremark](http://en.wikipedia.org/wiki/Futuremark)'s [3DMark06](http://en.wikipedia.org/wiki/3DMark#Versions) and [Vantage](http://en.wikipedia.org/wiki/3DMark#3DMark_Vantage) [benchmarking tools](http://en.wikipedia.org/wiki/Benchmark_(computing))[[31]](http://en.wikipedia.org/wiki/PhysX#cite_note-31)
* [Microsoft Robotics Studio](http://en.wikipedia.org/wiki/Microsoft_Robotics_Studio), an environment for robot control and simulation[[32]](http://en.wikipedia.org/wiki/PhysX#cite_note-32)
* [Nvidia](http://en.wikipedia.org/wiki/Nvidia) [SuperSonic Sled](http://en.wikipedia.org/wiki/Nvidia_demos" \l "SuperSonic_Sled" \o "Nvidia demos) and [Raging Rapids Ride](http://en.wikipedia.org/wiki/Nvidia_demos#Raging_Rapids_Ride), technology demos
* [OGRE](http://en.wikipedia.org/wiki/OGRE) (via the NxOgre wrapper), an [open source](http://en.wikipedia.org/wiki/Open_source_software) [rendering engine](http://en.wikipedia.org/wiki/Rendering_(computer_graphics))
* The [Physics Abstraction Layer](http://en.wikipedia.org/wiki/PAL_(software)), a physical simulation [API](http://en.wikipedia.org/wiki/Application_programming_interface) abstraction system (it provides [COLLADA](http://en.wikipedia.org/wiki/COLLADA) and [Scythe Physics Editor](http://en.wikipedia.org/wiki/Scythe_Physics_Editor) support for PhysX)[[33]](http://en.wikipedia.org/wiki/PhysX#cite_note-33)
* Rayfire, a plug-in for [Autodesk 3ds Max](http://en.wikipedia.org/wiki/Autodesk_3ds_Max) that allows fracturing and other physics simulations

# Comparatif

# Conclusion

# Réferences