

# Chapitre 1: Introduction à l'infographie

INF5071 — Infographie

Alexandre Blondin Massé

Université du Québec à Montréal

Hiver 2019

# Plan

- 1 Présentation du cours
- 2 Infographie 3D
- 3 Matériel et applications utiles

# Présentation du cours

# Informations générales

- **Trimestre:** Hiver 2019
- **Titre du cours:** Infographie
- **Sigle:** INF5071
- **Département:** Informatique
- **Enseignant:** Alexandre Blondin Massé
- **Courriel:** blondin\_masse.alexandre@uqam.ca
- **Site personnel:** <http://lacim.uqam.ca/~blondin>
- **Bureau:** PK-4525
- **Coordonnateur:** Alexandre Blondin Massé
- **Site du cours:** <http://lacim.uqam.ca/~blondin/fr/inf5071>
- **Plan de cours:** cliquer ici

## Description officielle (site de l'UQAM):

*« Connaître les algorithmes fondamentaux de l'infographie. Être capable de réaliser une application graphique simple à l'aide d'une bibliothèque graphique (par exemple, OpenGL). Rappels des notions de base en calcul vectoriel et en géométrie. Objets géométriques élémentaires. Codage des objets en deux et trois dimensions. Transformations linéaires en coordonnées homogènes. Composition de transformation. Transformation de perspective et modèles de représentation. Tracé de figures élémentaires. Représentation paramétrique des objets en trois dimensions (courbes de Bézier, B-splines). Fonctionnalités typiques offertes par les bibliothèques graphiques, par exemple, OpenGL. Autres sujets (selon le temps disponible) : modèle de lumière, espace de couleurs, surface de Bézier, lancer de rayons. »*

# Objectifs du cours

- Maîtriser les notions de base de l'**infographie**
- Savoir générer, modifier et manipuler des **images**
- Savoir générer, modifier et manipuler des **modèles** 3D
- Comprendre les notions de **géométrie** (2D et 3D) sous-jacentes
- Connaître certaines **structures de données** utilisées en infographie
- Connaître certains **algorithmes** utilisés en infographie
- Savoir les **implémenter**
- Développer des **applications graphiques** en 3D
- **Communiquer** à l'oral et à l'écrit ces différents aspects

# Modalités d'évaluation

## Projet de session (50%)

- Présentation du projet (5%)
- Présentation 1 (15%)
- Présentation 2 (20%)
- Remise finale (10%)

## Devoirs (20%)

- 2 devoirs
- 10% chacun

## Quiz (30%)

- 3 quiz
- 10% chacun

# Contenu détaillé

- 1 Introduction
  - 2 Images et géométrie 2D
  - 3 Modélisation et géométrie 3D
  - 4 Textures
  - 5 Animations
  - 6 Lumière
  - 7 Shaders
  - 8 Physique
  - 9 Réalité virtuelle?
- Autres sujets?

# Références

## Contenu du cours

- John Pile Jr. *2D Graphics Programming for Games*, Taylor & Francis Group, 2013.
- Eric Lengyel. *Mathematics for 3D Game Programming and Computer Graphics*, Course Technology PTR, 2012.
- Mike Bailey et Steve Cunningham. *Graphics Shaders: Theory and Practice*, CRC Press, 2012.
- Bernd Jähne. *Digital Image Processing*, Springer, 2005.

## Politiques de l'UQAM

- Règlement 18 sur la tricherie et l'intégrité académique (plagiat): <http://r18.uqam.ca/>.
- Politique 16 contre le harcèlement sexuel: [document pdf](#).

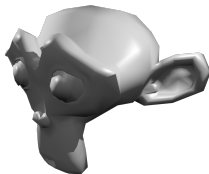
# Infographie 3D

# Maillage (*mesh*)

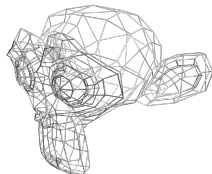
- Représentation **simplifiée** d'un objet 3D
- Aussi appelé **géométrie** d'un objet ou d'un modèle
- C'est un **graphe** plongé dans l'espace **euclidien**  $\mathbb{R}^3$
- Composé de **sommets** (*vertices*), **arêtes** (*edges*) et **faces**
- Qui stockent des **informations**: position, coordonnées de texture, couleurs, vecteurs normaux, etc.



sans lissage  
(*flat shading*)



avec lissage  
(*smooth shading*)

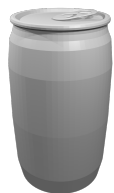


avec arêtes  
(*wireframed*)

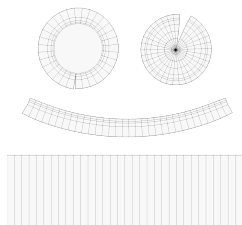
Suzanne, la mascotte de **Blender**

# Texture

- On peut **plaquer** une texture sur un **maillage**
- Pour cela, il faut **développer** le maillage
- Une texture est simplement une **image**
- Puis on en fait un **rendu**
- Les **coordonnées** de texture sont identifiées par  $u$  et  $v$



maillage  
(*mesh*)



développement  
(*UV-unwrap*)



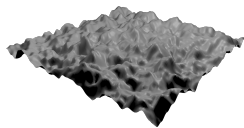
texture



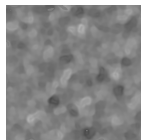
rendu  
(*rendering*)

# Images

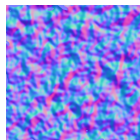
- Image = rectangle de pixels de dimensions  $w \times h$
- Les **images** jouent un rôle très important, même en 3D
- Elles peuvent représenter une **texture**
- Elles peuvent être le résultat d'une **projection** d'un objet 3D
- Mais elles permettent aussi d'**encoder** de l'information
- Deux principaux types de format: **vectorel** et **matriciel**
- L'**espace de couleurs** d'une image peut varier



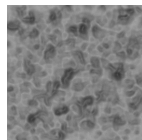
rendu  
(projection)



relief  
(*bump map*)

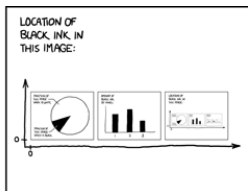
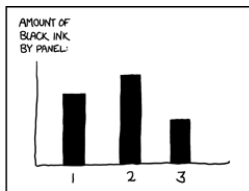
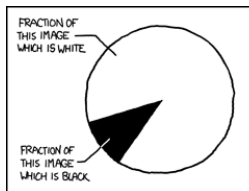


normales  
(*normal map*)

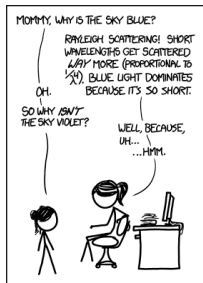


occlusion  
ambiante

# Images et couleurs



<https://www.xkcd.com/688/>

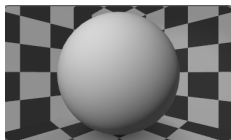


MY HOBBY: TEACHING TRICKY QUESTIONS TO THE CHILDREN OF MY SCIENTIST FRIENDS.

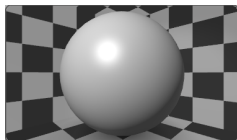
→ <https://www.xkcd.com/1145/>

# Lumière

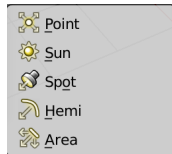
- Sans lumière, tout rendu d'une scène apparaîtrait en **noir**
- Il existe une multitude de **modèles** de lumière
- plus **simples**: Lambert, Phong, Blinn-Phong, ombrage de celluloïd, ...
- plus **complexes**: lancers de rayons
- Et des **phénomènes** complexes: réflexion, réfraction, caustiques, dispersion de lumière, illumination globale, ...
- Compromis entre **réalisme** et **temps de rendu**



Lambert  
(dans Blender)



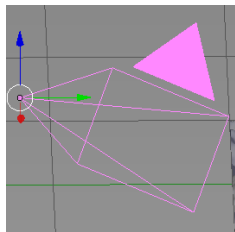
Blinn  
(dans Blender)



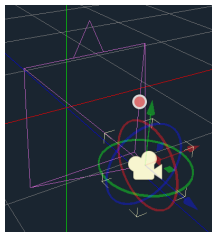
Types de lumière (*lamps*)  
disponibles dans Blender

# Caméra

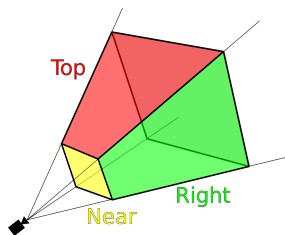
- Une scène peut contenir une ou plusieurs **caméras**
- C'est une **projection** de la scène selon une certaine vue
- **Types** de projection: perspective, orthographique, ...
- On peut lui appliquer des **transformations**
- Caractérisée par un **volume visible** (*view frustum*)
- Implique des notions d'algèbre **linéaire** et **matricielle**



caméra dans Blender



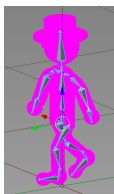
caméra dans Godot



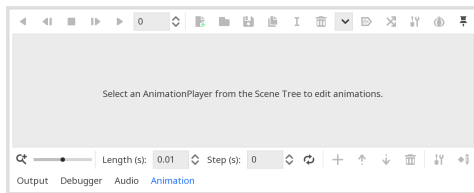
*view frustum*  
(source: [Wikipedia](#))

# Animation

- La plupart des objets d'une scène peuvent être **animés**
- Plusieurs **types** d'animations!
- **Propriétés**: position, rotation, couleur, transparence, ...
- **Déformations** *shape keys* dans Blender
- **Systemes articulés**: aussi appelé **squelette** ou **armature**
- **systemes de particules**: pas nécessairement animé
- **Phénomènes physiques**: vêtements, fluides, corps mous, ...



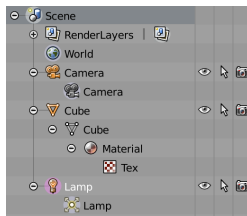
système articulé  
dans Blender



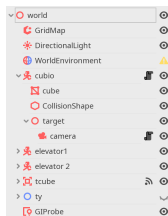
éditeur d'animation  
dans Godot

# Scène

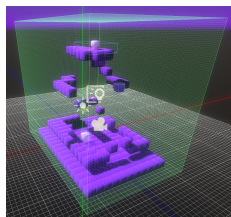
- Une **scène** est simplement un ensemble d'objets
- Elle est généralement organisée de façon **hiérarchique**
- Plus spécifiquement sous forme d'**arborescence** (*scene tree*)
- Plus il y a d'objets, plus le rendu est **complexe**
- Utilisation de **structures de données** (*octrees, kd-trees*)
- Et d'**algorithmes** efficaces (géométrie computationnelle)



scène dans Blender  
(arborescence)



scène dans Godot  
(arborescence)



scène complexe  
dans Godot

# Processeurs et cartes graphiques

CPU = *central processing unit*

- Là où s'effectuent les **calculs** en général
- Peu de tâches peuvent s'effectuer en **parallèle**
- Prépare les **données** à fournir au **processeur graphique**

GPU = *graphics processing unit*

- Fait partie de la **carte graphique**
- Hautement **parallèle**
- Limité dans le type de calculs: principalement **matriciel**
- Manipule des **structures de données** de bas niveau

## Shaders

- **Programmes** pour communiquer avec le GPU
- Plusieurs **types**: *pixel, vertex, geometry*, etc.

# Matériel et applications utiles

# Matériel

- Souris à 3 boutons (mollette) → **fortement recommandé** pour la modélisation et l'édition de scènes
- Tablette graphique → très **optionnel**, facilite la création et la manipulation d'images, ainsi que la sculpture
- Écran à stylet → encore plus **optionnel**



(source: [Coop UQAM](#))  
souris à 3 boutons  
et à mollette  
20-25\$



(source: [Wacom](#))  
tablette graphique  
(modèle Intuos)  
80-200\$ de base



(source: [Wacom](#))  
écran à stylet  
(modèle Cintiq Pro)  
≥ 800\$

**Note:** les produits Wacom sont présentés à titre d'exemple

# Logiciels

## Logiciel de dessin et/ou d'édition d'images

- Création d'**images**, application de **filtres**
- **Exemples**: Photoshop, Gimp, Krita, Inkscape

## Logiciel de modélisation 3D

- Logiciel de création de **modèles 3D**
- Pour créer des maillages, animations 3D, textures, etc.
- **Exemples**: Autodesk Maya, SketchUp, Blender

## Moteurs de jeu

- **Environnement** facilitant le développement d'un jeu vidéo
- **Exemples**: Unity, Unreal, Phaser, XNA, Phaser, Godot

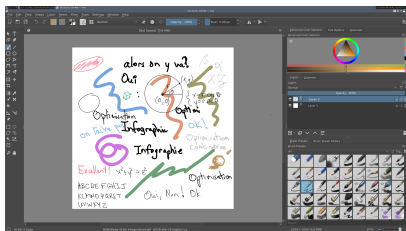
**Note**: dans le cours, je n'utiliserai que des **logiciels libres**

# Krita

- Développé par KDE (KDE = *Kool Desktop Environment*)
- Site officiel: <https://krita.org>
- License: GPLv3
- Principalement destiné aux **arts numériques**
- Permet aussi de transformer des **images**, p. ex. avec des filtres
- Système de **couches** (*layers*) vectorielles ou matricielles



Logo officiel de Krita  
(source: Krita)



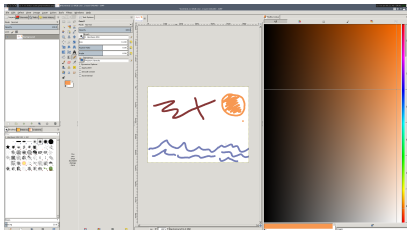
Interface de Krita (v.4.1.1)

# Gimp

- Développé par *the GIMP Development Team*
- Site officiel: <https://gimp.org>
- License: GPLv3
- Permet de **manipuler** et de **retoucher** des images
- Fournit un grand nombre de **filtres**
- Système de **couches** (*layers*)
- Comparable à Photoshop



Logo officiel de Gimp  
(source: [Gimp](#))



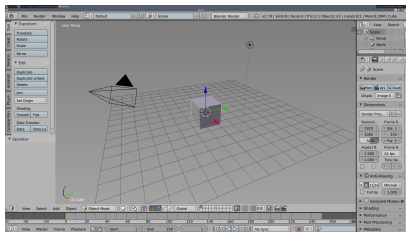
Interface de Gimp (v.4.1.1)  
avec [Awesome WM](#)

# Blender

- Développé par la Fondation Blender
- Site officiel: <https://www.blender.org>
- License: GPL
- Permet de créer des **modèles 3D** avec texture avancée
- Très **polyvalent**: sculpture, animation, physique, etc.
- Bonne **interaction** avec les moteurs de jeu
- Comparable à Autodesk Maya et Autodesk 3ds Max



Logo officiel de Blender  
(source: Blender)



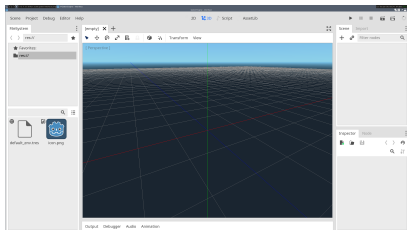
Interface de Blender  
(thème « 24X Blue »)

# Godot

- Initié par Juan Linietsky et Ariel Manzur
- Site officiel: <https://godotengine.org>
- License: MIT
- Permet de créer des jeux **2D** (mature) et **3D** (croissance)
- Chaque ressource graphique est une **scène**
- Bonne intégration avec Blender
- Comparable à Unity et Unreal



Logo officiel de Godot  
(source: [Godot](https://godotengine.org))



Interface de Godot, thème « Light »  
(v.3.0.6.stable.official.8314054)