

Cours de création de jeux vidéo

3. Art 2D et 3D Statique

par Panagiotis Tsiapkolis (Panthavma)

le 3 Mars 2024

– Art Visuel

- * On a vu comment programmer et concevoir un prototype
- * Aujourd'hui: aspect visuel direct, sans animations ou VFX.

— QUESTION:

Quelle structure de noeuds peut-on utiliser pour faire un perso de jeu de plateformes ?

- * Sprite → KinematicBody, CollisionShape
- * Sprite → KinematicBody → CollisionShape
- * KinematicBody → CollisionShape, Sprite
- * Spatial → KinematicBody, CollisionShape, Sprite

– REPONSE:

KinematicBody → CollisionShape, Sprite. On veut le corps physique comme racine, a qui on ajoute le collider. Le sprite reste attaché à la racine.

– QUESTION:

Comment on ajoute une accélération à une entité?

- * $position+ = accel$
- * $position+ = accel \times \delta_t$
- * $vitesse+ = accel$
- * $vitesse+ = accel \times \delta_t$

– REPONSE:

vitesse = *accel* \times δ_t . On peut vérifier avec l'analyse dimensionnelle:

$$m/s = m/s^2 \times s$$

2D

– Style graphique et Direction Artistique

- * Direction Artistique: Cohérence et style global
 - Sentiment Visé
 - Limitations Techniques
 - Limitations de Production

— QUESTION:

Comment est-ce qu'on utilise les couleurs ?

– Théorie des couleurs

- * Théorie des couleurs: qualités esthétiques et communicatives des couleurs
 - Définition: Teinte, Saturation, Valeur
 - Aspects esthétiques: Froid/Chaud, etc.
- * Combinaison: Relations entre couleurs (complémentaires...), composition

– Théorie des couleurs, conseils

- * Palettes simples: "Split Complementary", etc
- * Autre option: Choisir trois couleurs et leurs teintes intermédiaires
- * Outil: Passer en niveaux de gris pour voir où l'oeil est attiré

— Sprites

- * Sprite: Tableau 2D de pixels
- * Pixel: Couplet RGB(A) ou un indentifiant (Palettes Indexées)
- * Blit: Copie des pixels sur l'écran par rapport à un point fixe du pixel (ancre/origin)
- * Tailles souvent multiples de 2 pour la perf
- * Transformations: Rotation, Echelle, etc
 - Peut affecter le résultat (théorie du signal)

– Echelle et algorithmes



Source : Real Time Rendering

– Pixel Art

- * Style basé sur les limitations de la plateforme (exemple: Game Boy Color)
- * Resolution plus limitée (GBC: 160x144)
- * Couleurs limitées (56 parmi 32768) et par tile (4 sur une tile 8x8)
- * Sprites (8x8 / 8x16) vs Backgrounds, et leurs limitations
- * Autres restrictions (nb sprites par ligne / frame, mémoire...)

— QUESTION:

Est-ce que tu connais d'autres limitations et techniques pour les dépasser ?

– Vectoriel vs Raster

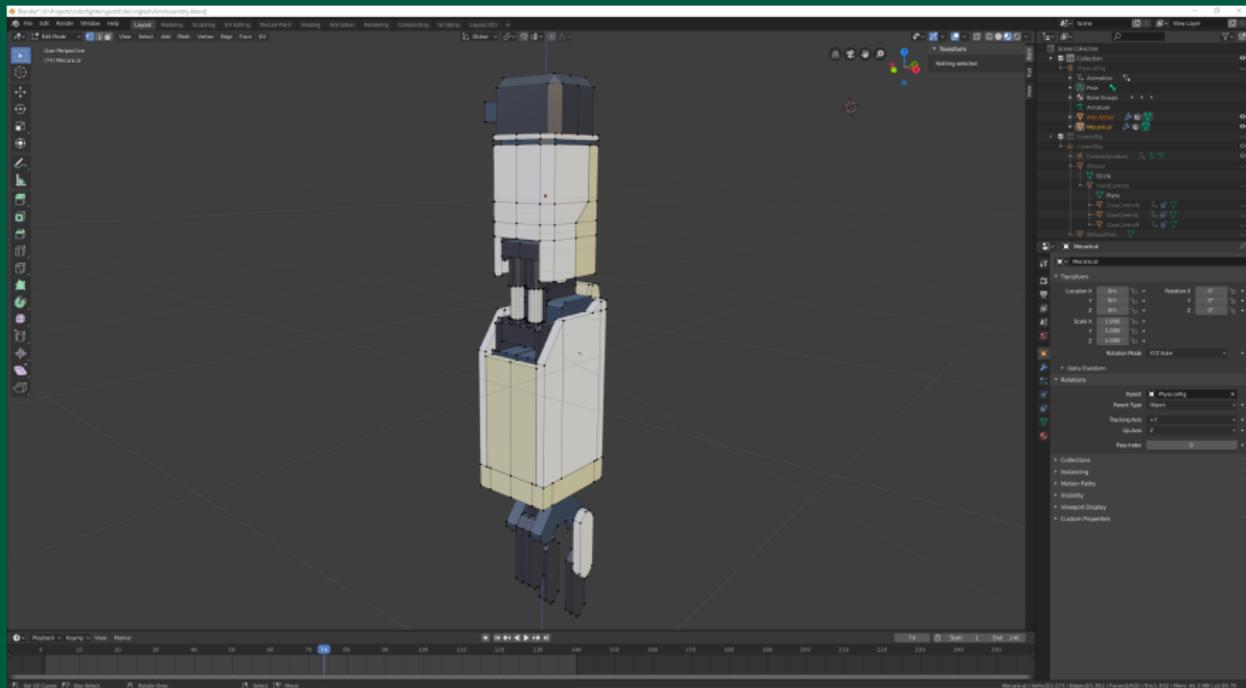
- * Vectoriel: Images définies par des formes mathématiques (primitives)
- * Avantage: Résolution parfaite
- * Désavantage: Temps de calcul

3D

— Modélisation 3D

- * Maillage (mesh): Graphe de sommets (vertex / vertices) assemblés en faces.
- * Matériau (material): Paramètres de rendu de surface et données associés (textures, etc)
- * Modèle 3D (model): Maillage + Matériaux + Autres données associées (rig, animations...)
- * Scène 3D: Modèles 3D + Lumières + Camera
- * On va utiliser un logiciel comme Blender pour placer les points

— Modélisation 3D



– Matériau et shaders

- * Calcul de la couleur du pixel est faite par un shader (programme GPU) selon le matériau (shader + paramètres)
- * PBR (Physically Based Rendering): Rendu basé sur la physique et paramètres mesurables
 - Albedo: Couleur de l'objet sous lumière blanche
 - Autres paramètres: Metalness (metal), Roughness (rugosité)...
- * Certains matériaux peuvent être complexes (réflexion, subsurface scattering...)

— Texture

- * Texture: Une image qu'on applique au maillage
- * Trois échelles de détail
 - Macro: Plusieurs pixels = Maillage
 - Micro: Sub 1px = Shader
 - Meso: 1-4px = Texture
- * Application via UV Mapping: chaque sommet du maillage associé à un point de la texture
- * Peut contenir de la couleur ou des paramètres
- * Beaucoup d'autres utilisations de textures: Tech Art

– Texture



Source : Real Time Rendering

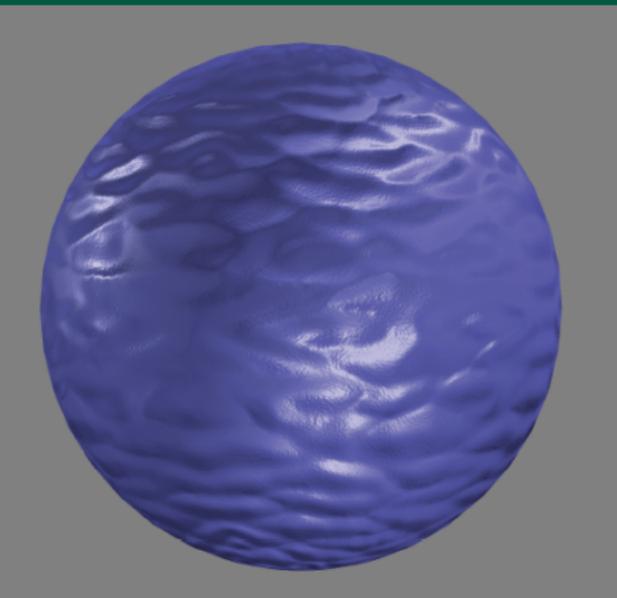
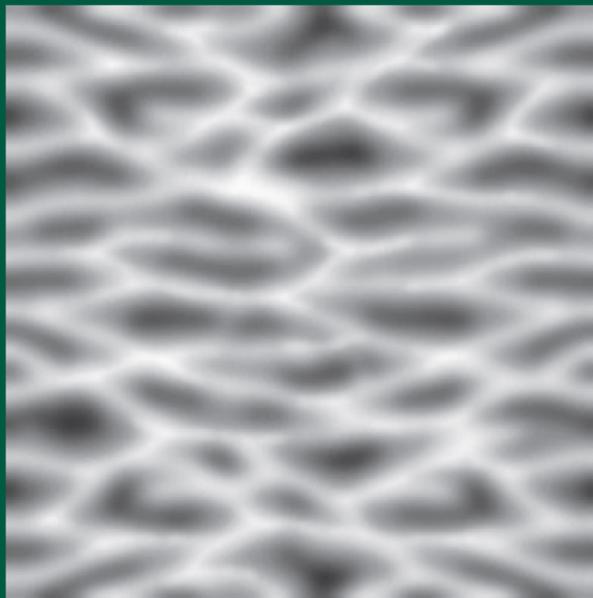
— QUESTION:

Est-ce que vous connaissez d'autres utilisations des textures?

– Bump mapping

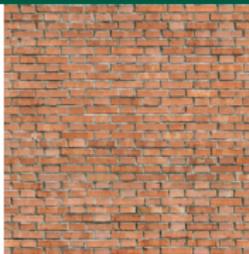
- * Exemple: Bump / Normal Mapping
- * Rajoute du faux détail géométrique pour l'éclairage.
- * Utilisation de texture de hauteur ou normales (direction de surface).
- * Limites aux angles rasants: on peut utiliser le déplacement mapping plutôt.
- * Souvent créés par projection depuis un model plus détaillé.

– Heightmap

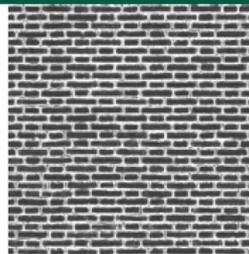


Source : Real Time Rendering

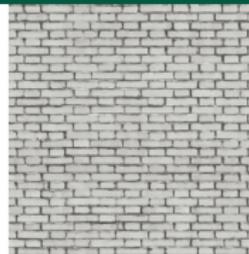
– Matériaux plus complexes



albedo
texture



roughness
texture



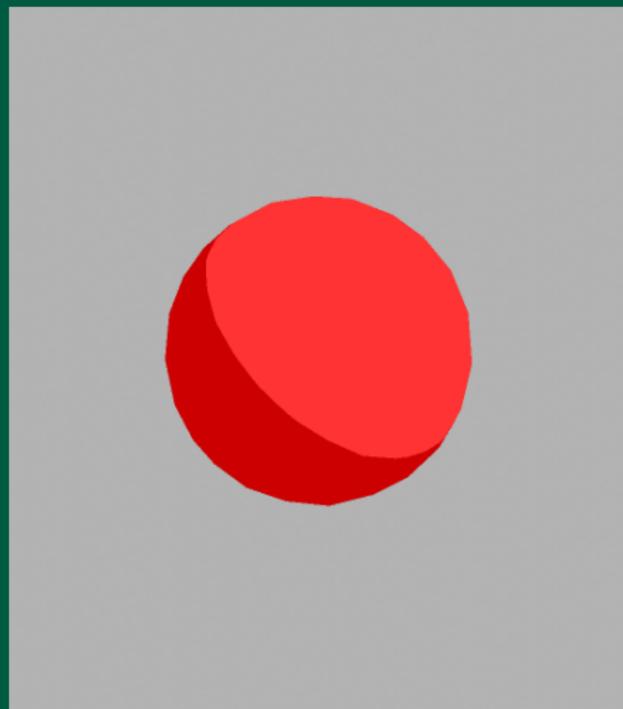
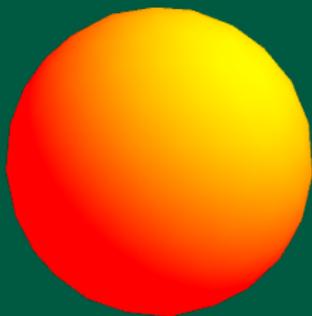
heightfield
texture

Source : Real Time Rendering

– Toon Shading

- * Ombrage stylisé: réduire l'ombrage à quelques tons
- * Besoin d'une fonction pour le faire (souvent dépendant d'un coefficient de lumière)
- * Toon Ramp: Façon de stocker cette fonction dans une texture
- * On parlera plus de ce genre de point dans les sujets avancés

– Matériaux plus complexes



Source : Panthavma

– Skeleton / Rigging

- * Note additionnelle: Animation par squelettes (existe en 2D aussi)
- * Squelette: Hierarchie d'os qui sert à déplacer le model comme une poupée
- * Weight Painting: Association des sommets avec les os
- * Rig: Squelette + Controles

Intégration dans des jeux

– Pipeline

- * Description des étapes et outils pour faire un personnage / objet / etc.
 - Concept Artist fait des dessins (en particulier dessins de référence)
 - Création du maillage high poly via sculpting
 - Création du maillage low poly et projection via baking
 - Création des matériaux et textures
 - Création du rig et weight painting
 - Création des animations

– Pipeline 2D



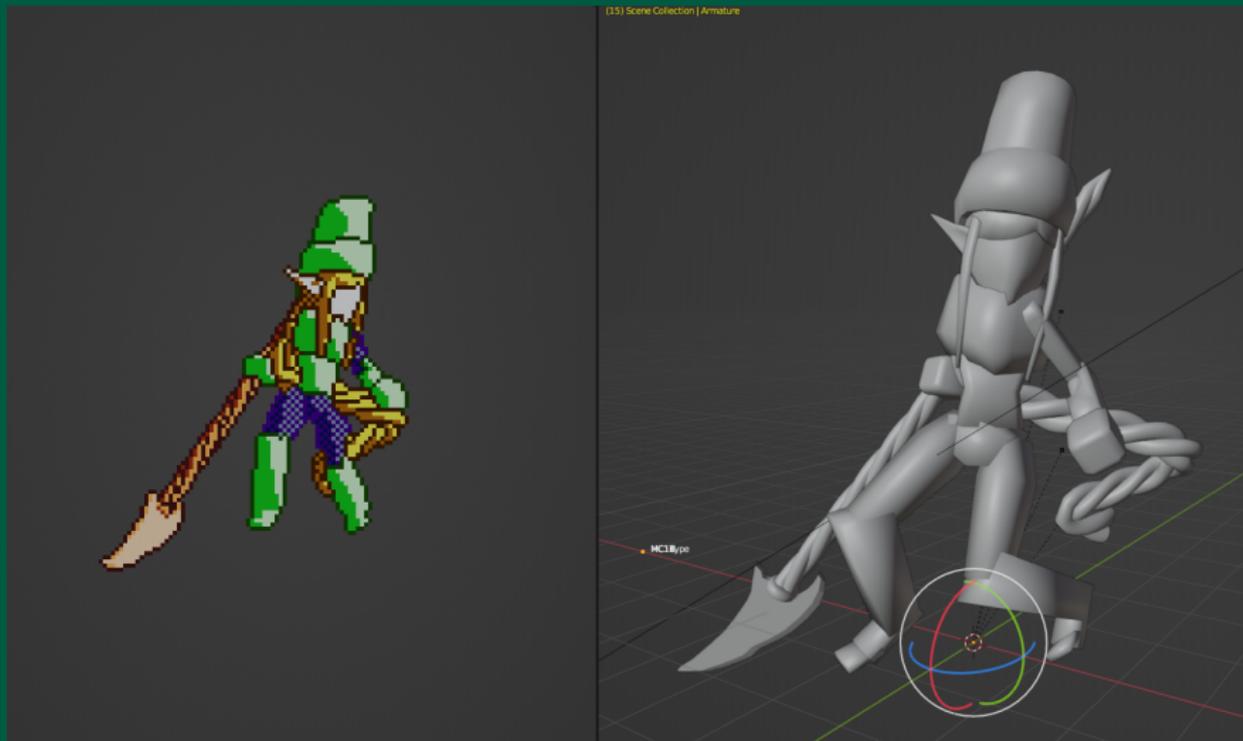
Source: BlazBlue

– Pipeline 3D 1



Source: Dead Cells

– Pipeline 3D 2



Source: Molten Winds

– Packs d'art, styles peu intenses

- * Façons d'éviter si c'est pas ton truc
- * Utiliser un pack d'art (KennyNL, OpenGameArt...): Cohérence peut être un problème, bien aussi comme base pour éditer
- * Styles moins complexes: Terminal (ncurse), Pixel Art basse res, Low Poly

— Questions ?

- * Lien Discord : <https://discord.gg/CWjWfC9K9T>
- * Site du cours : <https://panthavma.com/gamedevclass/fr/>
- * La prochaine fois : Game design, mécaniques et émotions (10 Mars 2024)
- * A côté : Tester l'art 2D et 3D! Définir une direction artistique!
- * Logiciels: Blender, GIMP, Inkscape, Krita