Modélisation géo -> Remanier des formes assez simples.

Triangulation et subdivision de mech.

Monde virtuel en fin d’année -> Récupération de donnée géo spatiale pour ensuite l’introduire à Unity puis faire de la génération procédurale de route / ville.

Chapitre 1 : De l’acquisition à la production d’une image fixe ou animée.

Plan découpé en triangle puis raffiné par tout un processus que l’on va aborder.

Grande place dans les jeux vidéo, le cinéma, la pub … Mais aussi le jeu sérieux pour toutes les problématiques de simulation. (Représentation 3D et architectural, médecine).

Studio de flight Simulator qui propose parfois des stages avec un test d’entrée en C++

Check: <https://www.asobostudio.com/fr/carriere>

On vise souvent un compromise entre le réalisme / Temps Réel.

Exemple : Incendie, très compliqué en temps réel sans que l’ordinateur n’en souffre.

Place et rôle du mesh (modèle) :

LOD en fonction de la distance. Information stockée dans le modèle.

On va souvent lier les modèles, les rendre inter opérable pour mieux qu’ils interagissent entre eux.

Pour vérifier la qualité des modèles, on peut faire de la comparaison entre les calculs de 2009 et 2018 (exemple de dates). Ensuite il faut comprendre pourquoi il y a ces changements.

Une bonne pratique est de questionner la qualité / légitimité de notre donnée.

Modélisation géométrique :

Utilisation de points, faces.

Utilisation de modèles Cubes, Triangles.

Tout ce qui est lié à la création de maillage.

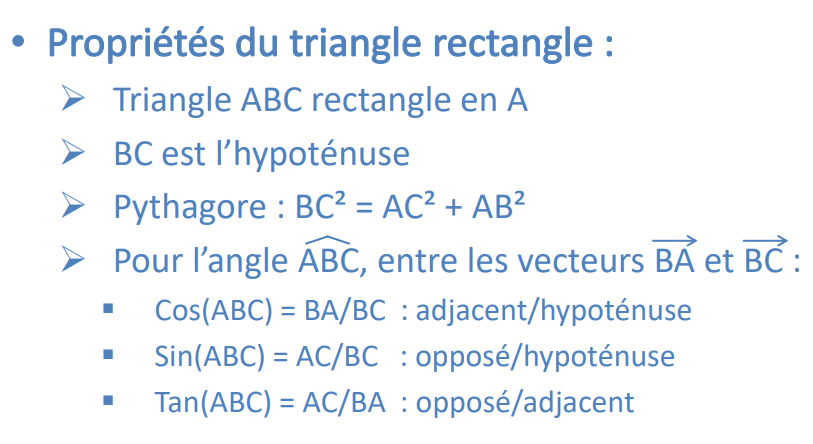
Habillage couleur – Cours de L3 à Montpellier en fait.

Habillage de texture.

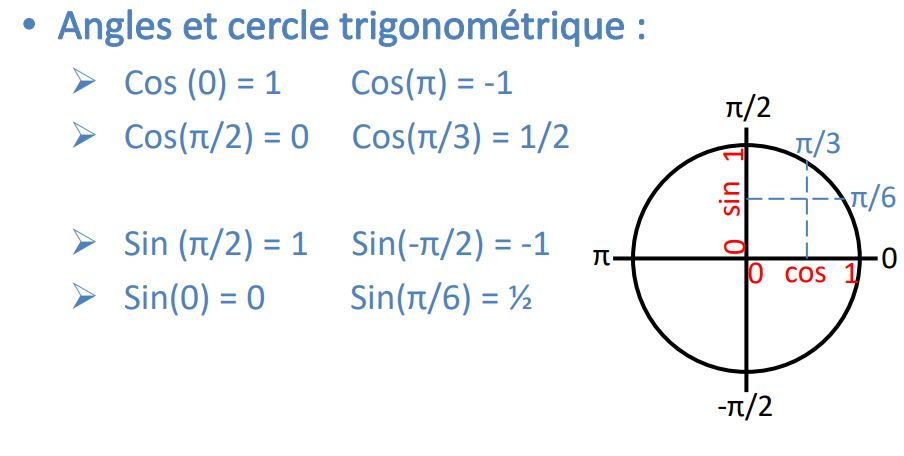
2 types d’illumination -> Globale (dépendant du maillage) / locale (comment l’objet réagi).

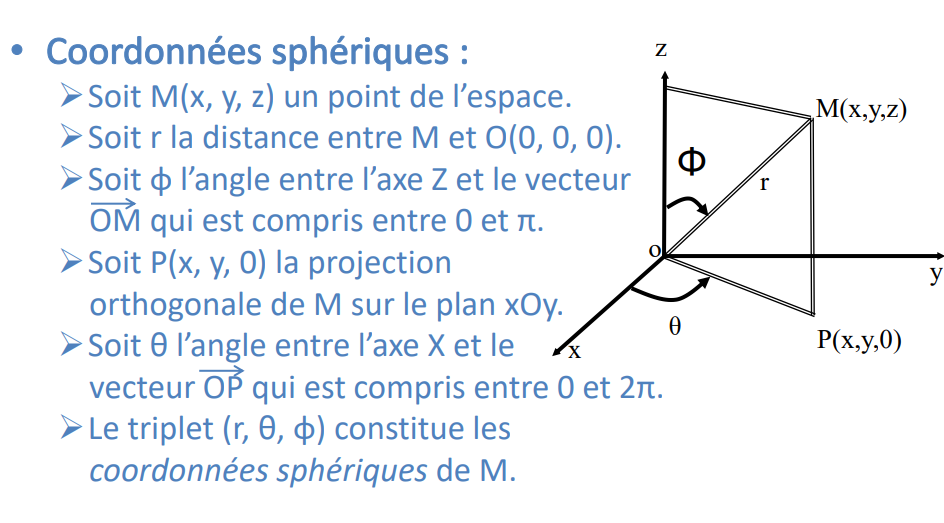
Représentation Surfacique polyèdres et quadratiques :

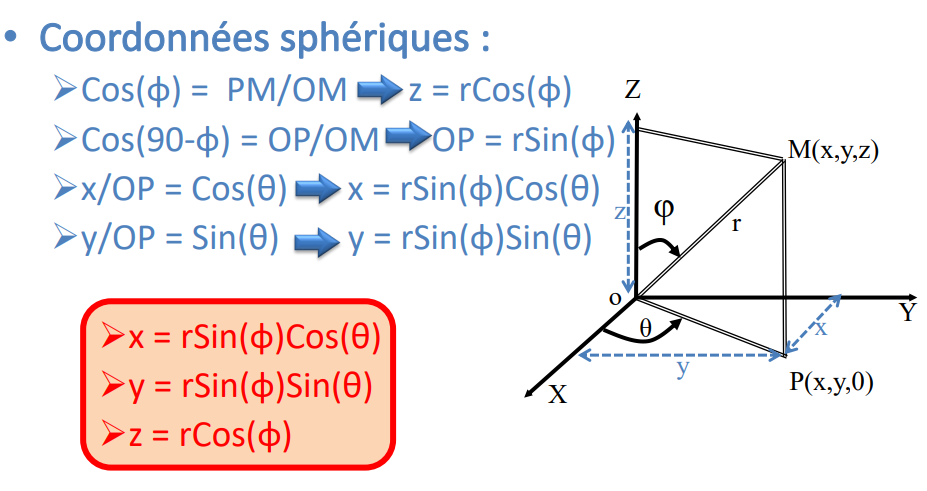
Représentation surfacique -> Surface extérieur d’un modèle.



Rappel pour la suite (important).







Cos(phi) = z / r donc z = r\*cos(phi)

Cos(90-phi) = OP / OM donc OP = r\*Sin(phi)

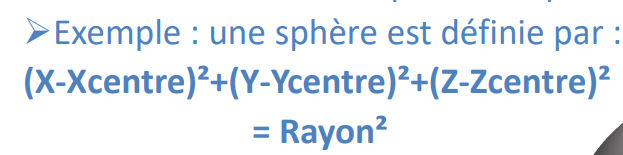
…

Polyèdre =/= surface continue

Un Polyèdre est défini par un ensemble de sommets et de faces.

Bien sûr, faire attention à l’ordre d’initialisation des sommets pour la déclaration des normales.

(Sur unity : recalculate normals).



On peut ainsi définir la surface par autant de point que l’on veut et n’importe où sur la surface (contrairement au polyèdre).

Quadrique : La classe de surface quadriques contient les cylindres, les cônes, les sphères …

Une quadrique a une équation implicite de degré 2 de la forme F(x, y, z) = 0 avec

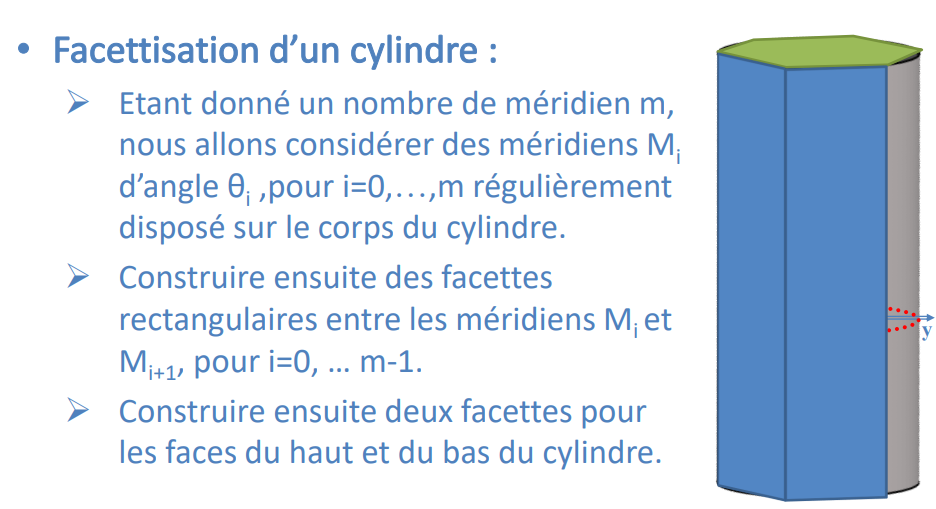
F(x, y, z) = Ax² + 2Bxy + 2Cxz + 2Dx + Ey² + 2Fyz + 2Gy + Hz² + 2Iz + J

Le cylindre :

A pour équation x² + y² = r²

Sa hauteur est définie par un réel positif : h

Les méridiens sur un cylindre de révolution de rayon r et de hauteur h sont les segments de droites contenus dans le corps du cylindre, de longueur h, parallèles à l’axe du cylindre.



Le cône :

A pour équation : (z – z sommet) ² = z sommet² / r² \* (x² + y²)

Un méridien -> Découpe du mesh verticales servant aussi à définir sa « résolution ».

Un parallèle -> Equivalent à un méridien mais horizontal.

Les deux combinés peuvent servir à réaliser des sphères.

Les cylindres par exemple n’utilisent que des méridiens pour leurs définition.