



# La 3D temps réel avec OpenGL

Mise à jour : Fév. 2024

**Durée** : 5 jours - 35 heures

## OBJECTIFS PÉDAGOGIQUES

- Appréhender l'API et les concepts importants d'OpenGL, ainsi que les particularités de la 3D temps réel, autant avec le pipeline fixe qu'avec les shaders
- Découvrir l'étendu des fonctionnalités d'OpenGL.

## PRÉREQUIS

- Compétences en langage C, quelques notions concernant le monde de la 3D.

## PARTICIPANTS

- Développeur.

## MOYENS PÉDAGOGIQUES

- Réflexion de groupe et apports théoriques du formateur
- Travail d'échange avec les participants sous forme de réunion-discussion
- Utilisation de cas concrets issus de l'expérience professionnelle
- Validation des acquis par des questionnaires, des tests d'évaluation, des mises en situation et des jeux pédagogiques
- Remise d'un support de cours

## MODALITÉS D'ÉVALUATION

- Feuille de présence signée en demi-journée
- Evaluation des acquis tout au long de la formation
- Questionnaire de satisfaction
- Attestation de stage à chaque apprenant
- Positionnement préalable oral ou écrit
- Evaluation formative tout au long de la formation
- Evaluation sommative faite par le formateur ou à l'aide des certifications disponibles

## MOYENS TECHNIQUES EN PRÉSENTIEL

- Accueil des stagiaires dans une salle dédiée à la formation, équipée d'ordinateurs, d'un vidéo projecteur d'un tableau blanc et de paperboard

## MOYENS TECHNIQUES DES CLASSES À DISTANCE

- A l'aide d'un logiciel comme Teams, Zoom etc... un micro et éventuellement une caméra pour l'apprenant,
- suivez une formation en temps réel et entièrement à distance. Lors de la classe en ligne, les apprenants interagissent et communiquent entre eux et avec le formateur
- Les formations en distanciel sont organisées en Inter-Entreprises comme en Intra-Entreprise
- L'accès à l'environnement d'apprentissage (support de cours, labs) ainsi qu'aux preuves de suivi et d'assiduité (émargement, évaluation) est assuré
- Les participants reçoivent une invitation avec un lien de connexion
- Pour toute question avant et pendant le parcours, une assistance technique et pédagogique est à disposition auprès de notre équipe par téléphone au 03 25 80 08 64 ou par mail à [secretariat@feep-entreprises.fr](mailto:secretariat@feep-entreprises.fr)

## ORGANISATION

- Les cours ont lieu de 9h à 12h30 et de 13h30 à 17h

## PROFIL FORMATEUR

- Nos formateurs sont des experts dans leurs domaines d'intervention
- Leur expérience de terrain et leurs qualités pédagogiques constituent un gage de qualité

## ACCESSIBILITÉ

- Les personnes atteintes de handicap souhaitant suivre cette formation sont invitées à nous contacter directement, afin d'étudier ensemble les possibilités de suivre la formation. Notre organisme peut vous offrir des possibilités d'adaptation et/ou de compensations spécifiques si elles sont nécessaires à l'amélioration de vos apprentissages

# Programme de formation

## Présentation

- versions et historique (1.x à 4.x, ES1, ES2)
- place d'OpenGL sur le marché actuel de la 3D
- principes de fonctionnement d'une carte 3D
- pipeline fixe et pipeline programmable
- extensions OpenGL
- bindings et langages

## Initialisation et contexte

- création de l'espace de rendu
- les API concernées : GLX, WGL, CGL, EGL, ...
- les abstractions possibles : GLUT, SDL, ...
- gestion des extensions (GLEW, GLEE, ...)
- le cas de l'API GLU

## Principes de base

- définition d'une scène dans un espace en 3D
- états de la machine OpenGL
- espace de visualisation : Frustum

## Formes, volumes et géométries

- points, lignes et polygones
- concepts : les surfaces évaluées (Bézier) et les NURBS de GLU
- géométries arbitraires
- performances et triangles
- mode immédiat, listes d'affichages, Vertex Array, VertexBuffers

## Matrices

- Rôle des matrices de la machine OpenGL
- Matrice de visualisation
- Matrice de transformation
- Rotations, translations

## Eclairage

- Rôle et fonctionnement de l'éclairage
- Simplifications du modèle d'éclairage
- Mise en place et définitions
- Déplacements des sources lumineuses
- Gestion des couleurs
- Gestion des matériaux
- Les normales (déduction et lissage)
- Les spots

## Le blending et les transparences

- intérêts et problématique du blending
- problématiques des superpositions blendées

## Application de textures

- Principes du texturage
- Chargement de textures
- Mise en place de coordonnées de texture
- Filtrages (linéaires, bilinéaires)
- MipMapping
- Matrice de texturage
- Extensions (multitexturing, textures 3D, ...)
- Précisions sur le blending de textures

## Tampons

- Tampon de profondeur (Z-buffer)
- Tampon d'accumulation
- Tampon "pochoir" (stencil buffer)
- Framebuffer Objects (FBO)
- Utilisations avancées des tampons (réflexions, blur, stencil shadows, cell shading, ...)

## Shaders

- présentation
- Vertex Shaders et Fragment Shaders
- Geometry Shaders (OpenGL 3.2) et tessellation (4.0)
- compilation et édition des liens des shaders
- le langage GLSL
- types, passages d'arguments, ...
- branchements et itérations
- mise en œuvre (toon shaders, normal mapping, post-processing, ...)

## Réalisme d'une scène

- ombres
- gestion du brouillard
- antialiasing
- skyboxes, dômes, ...
- particules et impostors
- gestion temporelle
- textures animées
- physique d'un environnement 3D
- textures animées, render-to-texture (RTT)
- gestion des entrées utilisateur
- workflow de création et gestion des assets
- performance et mémoire

## Présentation du GPGPU

- concepts de calcul embarqué dans le GPU
- intérêts et contraintes
- Shaders et FBO
- OpenCL (ouvert)
- CUDA (NVidia)