

Présentation des Projets

Année 2023-2024

[Les projets proposés]

- Comic Book Reader
 - Une liseuse de bandes dessinées
- TETRIS NG
 - Un jeu de TETRIS modernisé
- Object Oriented Ray Tracer
 - Un Ray Trace orienté objet modulaire et extensible.

[Comic Book Reader (CBR) - 1]

- Réalisation d'une liseuse prenant en charge les formats des BDs
- Gestion a minima des formats .CBR et format .CBZ.
- Amélioration du Rendu Graphique
Confort de lecture
Réactivité : (masquer le temps pris par les différentes tâches, en anticipant la lecture et le redimensionnement des pages, ...)

[Comic Book Reader (CBR) - 2]

- Caractérisation de CBR
 - Formats d'image (.png, .jpg, .jp2, etc) extensibles
 - Formats de conteneurs extensibles
- Extension
 - Format .pdf/.epub

[Comic Book Reader (CBR) - 3]

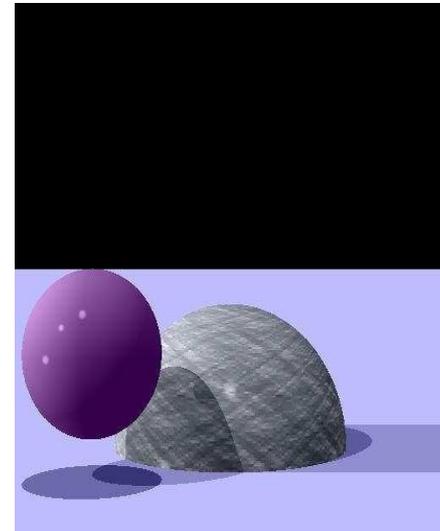
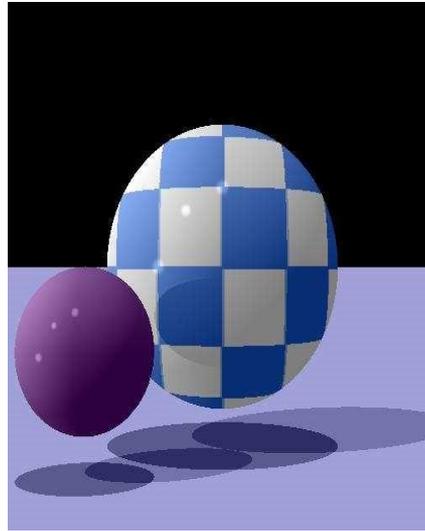
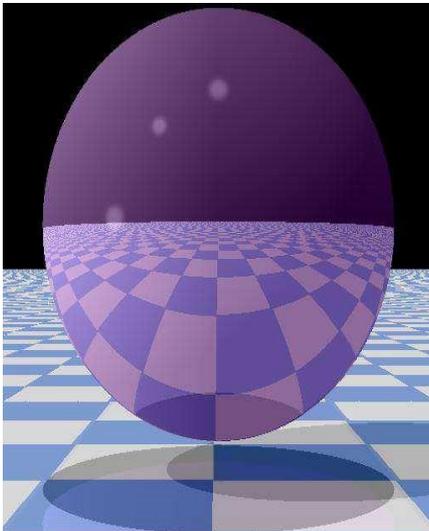
- Les point différenciants
 - Qualité de l'inteface graphique, possibilité du tactile mais aussi souris & clavier
 - Facilité d'usage
 - Quid d'une gestion de bibliothèques?

Object Oriented Ray Tracer (OORT) - 1

- Rendu performant de scènes complexes par lancer de rayon
- Support d'un modèle de description des objets d'une scène extensif
 - Format de spécification des objets
 - Bibliothèque de manipulation des objets

Object Oriented Ray Tracer (OORT) - 2

- Proposer une architecture permettant d'implanter plusieurs moteurs de rendu d'image par lancer de rayons
- Paralléliser le calcul pour arriver à du quasi temp-réel



Object Oriented Ray Tracer (OORT) - 2

- Les points différenciants
 - Comment décrire la scène, langage ?
Input d'un autre logiciel comme blender ?
 - La qualité du rendu, atténuation des différents défauts de rendu
 - La vitesse du rendu
 - L'extensivité soit au niveau du moteur soit au niveau des objets et textures.

[TETRIS NG]



- Implantation de TETRIS
 - Soit en mode multi-joueurs (entre plusieurs ordinateurs),
 - Soit en mode 3D,
 - Eventuellement les deux

Les aspects différenciant entre les projets (1)

- Comic Book Reader
 - Interface Graphique
 - Manipulation d'images
 - Gestion des formats
 - Filtrages
 - Gestions de fichiers compressés (Zip, Rar)
 - Confort d'utilisation par programmation asynchrone.

Les aspects différenciant entre les projets (2)

■ Tetris NG

- Interface Graphique temps réels
- Génération des briques graphiques 3D
- Distribution sur plusieurs machines
- Confort d'utilisation par programmation synchrone (prise en compte immédiate des commandes même en réseau)

Les aspects différenciant entre les projets (3)

- Object Oriented Ray Tracer
 - Bibliothèque de représentation des objets constituant une scène (langage de représentation)
 - Architecture unifié pour plusieurs moteurs de rendu de scène
 - Augmentation de la performance par parallélisation
 - Richesse et qualité du rendu de scène.

Méthodologie commune à tous les projets

- Phase 1 : Analyse
- Phase 2 : Implantation d'un sous-ensemble des fonctionnalités
- Phase 3: Ajout des fonctionnalités complémentaires en profitant de la modularité offerte par la programmation orientée objet.

[Analyse du projet]

- Déterminer les fonctions essentielles à réaliser
 - Exemple CBR : permettre de feuilleter des images stockées dans un livre dont la qualité a été améliorée pour les rendre plus lisibles
 - Chargement des images à partir d'un format propre au livre,
 - Traitement des images par filtrage
 - Affichage dynamique asynchrone
 - Mise en cache
 - ...

Implantation d'un sous-ensemble des fonctionnalités

- Déterminer les fonctions essentielles à réaliser
 - Exemple OORT : permettre le rendu de scène complexe en quasi temps réel.
 - Définition d'un format de description de scène riche et extensible
 - Objets, sources lumineuses, textures, ...
 - Fourniture d'une bibliothèque d'objets modulaires et extensives.
 - Parallélisation/Distribution (sur plusieurs machines) pour augmenter la performance de calcul
 - Possibilité d'utiliser différents moteurs de rendu d'image
 - ...

Implantation d'un sous-ensemble des fonctionnalités

- Implanter un sous-ensemble des fonctions de base pour l'applicatif
- **Exemple TETRIS**
 - Mode mono-joueur
 - Un seul plateau de jeu
 - Les objets simples de TETRIS (en 2D)

Implantation d'un sous-ensemble des fonctionnalités

■ Exemple OORT

- Un ensemble d'objets simples (Plan, Shère, Cube, ...)
- Surface uniforme (non texturée)
- Un moteur de calcul non optimisé

■ Exemple CBR

- Lecture d'un livre composé d'image .jpg.
- Affichage des images sans filtrage
- Fonctions de lectures de base d'une liseuse (première page, dernière, page suivante, précédent)

Ajout des fonctionnalités complémentaires

- Introduire de nouvelles classes dérivées permettant de supporter :
 - De nouveaux objets (OORT)
 - De nouveaux types d'image (CBR)
 - De nouveaux plateaux ou de nouvelles pièces (TETRIS), par exemple en 3D

Ajout des fonctionnalités complémentaires

- Introduire de nouveaux algorithmes
 - Nouveaux algorithmes de filtrage d'image (CBR) pour augmenter le contraste, la lisibilité du texte, corriger les couleurs....
 - Nouveaux algorithmes de rendu d'image pour OORT
 - Nouveaux algorithmes gérant les plateaux de jeu pour TETRIS
 - Mode « Pousser » de TETRIS DS
 - Mode multi-joueurs..

Ajout des fonctionnalités complémentaires

- Améliorer les performances
 - Masquer les latences en faisant de la programmation asynchrone (CBR)
 - Augmenter la performance en parallélisant (OORT)

Ajout des fonctionnalités complémentaires

- Améliorer l'expérience de l'utilisateur
 - Ajouter des fonctionnalités à la lecture des livres (gestion des bibliothèques de livres, extraction de pages,...)
 - Ajouter des nouveaux mode de jeux pour TETRIS
 - Mode distribué avec un serveur et deux ou plusieurs consoles de jeux
 - Gestion des scores de manière centralisé

[NE PAS REINVENTER]

- S'appuyer sur des bibliothèques pour les fonctions comme par exemple:
 - Lire/Ecrire des fichiers XML, JSON, etc
 - Lire/Ecrire des fichiers compressés,
 - Manipuler des images, décoder et encoder les différents formats d'images
 - Effectuer des traitements d'image
 - Communiquer sur le réseau...

[L'interface Graphique]

- S'appuyer sur une bibliothèque disponible
 - QT
 - wxWidget
 - SML
 - SFML
 - WinUI 3.0
 - ...

Modalités de réalisation du projet

- Seul ou en binôme
- Date de soutenance :
 - Début mars 2023.
 - Choix des dates avant la fin d'année.
- Soutenance :
 - Présentation de 10 minutes, démonstrations et questions

Critère d'évaluation du projet

- Qualité de l'architecture du programme
 - Bonne séparation des différentes fonctions
 - Interaction entre les différents blocs pertinents.

Critère d'évaluation du projet

- Démonstration de l'utilisation pertinente des constructions objets
 - Modularité, extensivité
 - Factorisation du code
 - Lisibilité du code

Critère d'évaluation du projet

- Aboutissement du logiciel développé
 - Bon fonctionnement de ce qui a été implanté
 - Extensibilité du logiciel
 - Bon niveau de performance
- En bonus,
 - La qualité du rendu graphique par exemple,
 - Le nombre de fonction
 - La qualité de l'interface graphique,
 - ...

Conclusion

- Choix du projet
 - (Par email à mon attention bruno.monsuez@ensta-paris.fr)
 - Descriptifs des projets présents sur le Wiki des cours:
<http://perso.ensta.fr/~bmonsuez/Cours>
- Conseil
 - Choisir un sujet qui vous plait (vous y passerez de nombreuses heures)
 - Les séances à partir de décembre sont là pour vous aider dans la réalisation des projets
 - Aucun projet n'est plus simple ou plus compliqué.