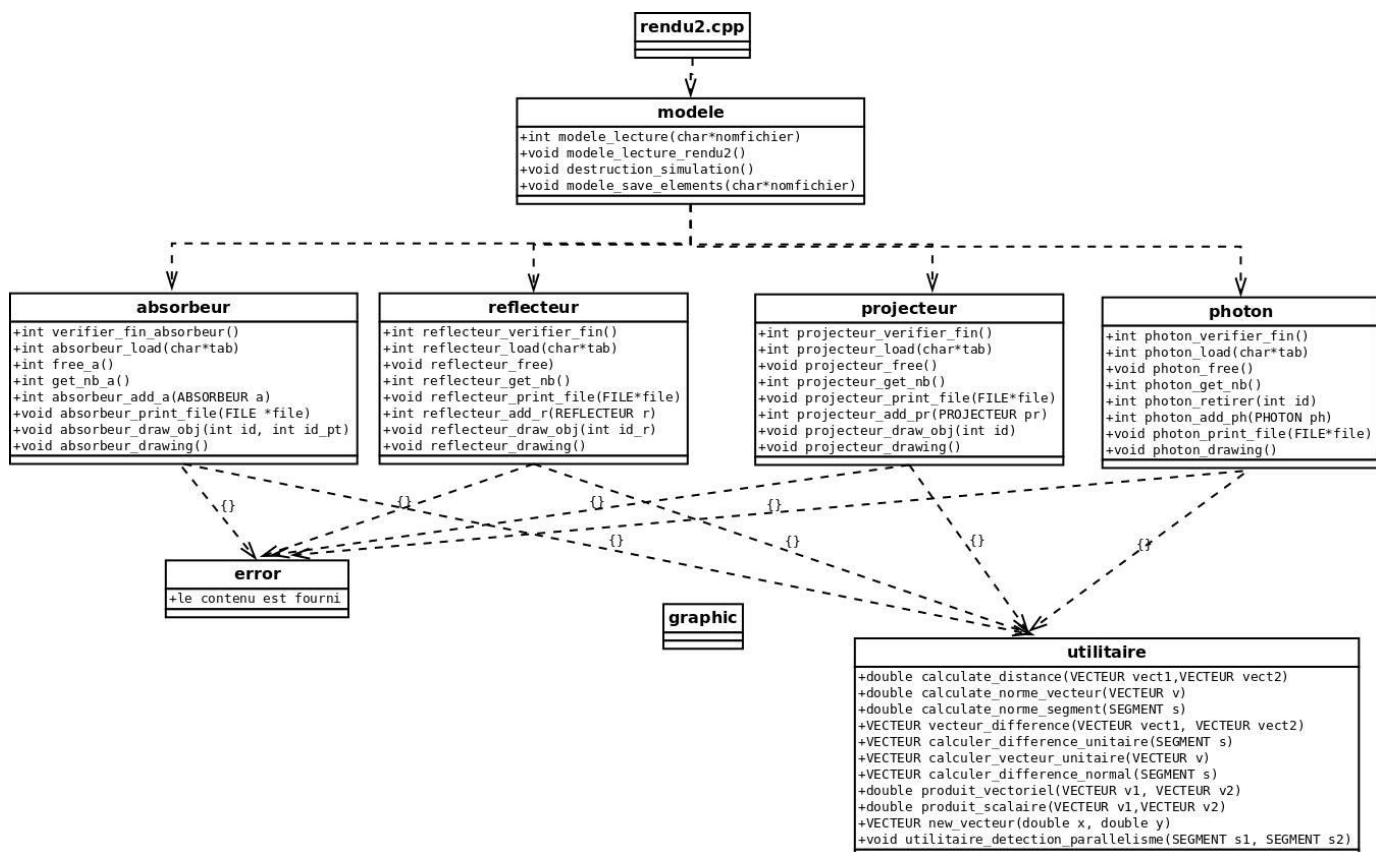


ARCHITECTURE DU RENDU FINAL

L'architecture de la figure 11b. est pour le moment respectée. Cependant, il est possible qu'elle soit légèrement modifiée d'ici le rendu final en raison de la mise en place de l'animation.

DESCRIPTION DES STRUCTURES DE DONNEES POUR LE RENDU FINAL

La première modification apportée comparativement au rendu 1 est la mise en place des tableaux dynamiques (pour les modules projecteur, absorbeur et reflecteur) et de listes chaînées (pour les photons) avec les allocations dynamiques de mémoire, modifiant ainsi dans chacun des modules correspondant les fonctions permettant la création de nouveaux objets. Cette modification a également engendré la mise en place de fonction « destructeur » pour libérer la mémoire lorsque celle-ci n'est plus requise. Dans chacun des modules, des fonctions de dessin ont également été écrites pour le dessin dans la fenêtre OPEN GL. A cela s'ajoute également l'écriture de la fonction `modele_verification_rendu2()`, avec l'écriture dans le module utilitaire de toutes les fonctions mathématiques nécessaires à la vérification des intersections et superpositions, ainsi que l'écriture de la fonction de sauvegarde.



LISTE DES FONCTIONS DU MODULE MODELE APPELEES DANS LE MODULE MAIN

Les fonctions `modele_lecture(char *nomfichier)` et `modele_verification_rendu2()` qui, comme décrit dans le sujet, servent respectivement à lire le fichier texte d'entrée et à vérifier les problèmes d'intersection et de superposition des objets, sont toutes deux appelées dans le module main (pour notre projet ce module est appelé rendu2). La fonction `destruction_simulation()`, qui permet la destruction totale du contexte de simulation par libération de la mémoire, est également appelée dans ce même module. Enfin, la dernière fonction du module modele appelée dans le module rendu2 est la fonction `modele_save_elements(char *nomfichier)`, cette dernière permettant l'écriture d'un fichier texte lorsque l'on appuie sur le bouton save.