

QUELQUES APPROCHES POUR CONCEVOIR UN MOTEUR

Problème

- ▶ En cas d'évolution faire remonter des propriétés communes à la classe parente commune (il faut qu'elle existant pas trop loin dans la hiérarchie)
- ▶ Risque d'avoir de grosses classes qui font plusieurs choses et qui ne respectent pas le principe de responsabilité unique de SOLID (« Single Responsibility Principle » SRP)
- ▶ Modifier une classe ne respecte pas le principe « ouvert/fermé » de SOLID (« Open/Close »)

QUELQUES APPROCHES POUR CONCEVOIR UN MOTEUR

- ▶ Approche par composition ou Modèle Entité-Composant
- ▶ Le comportement des entités est défini dans des composants (des objets) définis en dehors de l'entité
- ▶ L'entité contient une liste de composants
- ▶ Les composants vont à chaque image effectuer le traitement qu'ils implémentent
- ▶ Exemple : afficher l'entité, la déplacer, vérifier des collisions, etc.