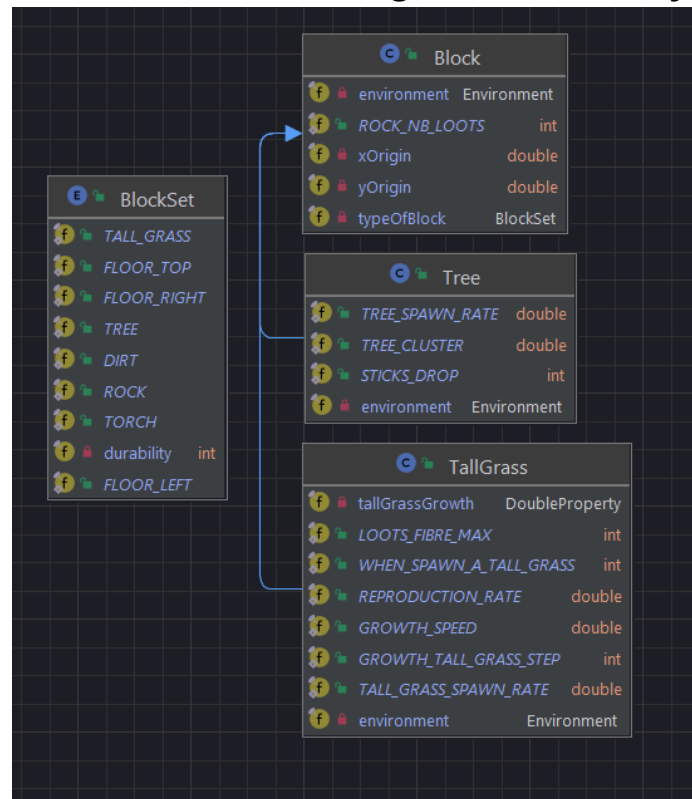


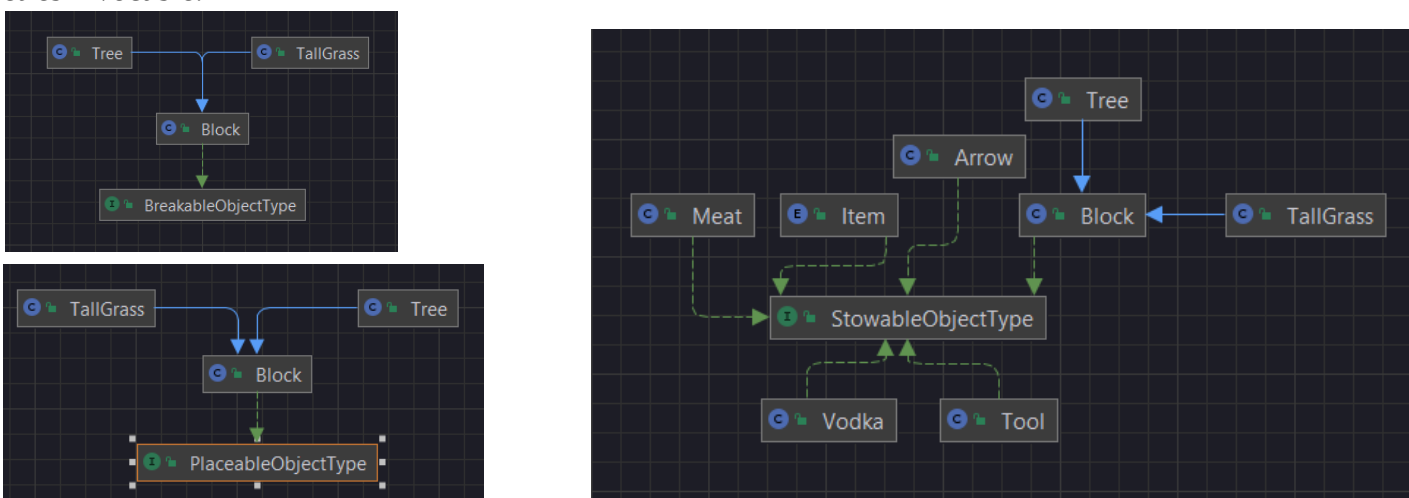
## Vue d'ensemble des classes gèrent les blocs du jeu



Ce diagramme permet de voir tous les blocs disponible grâce à l'enum, et mettre en avant ceux qui ont des propriétés propres à eux.

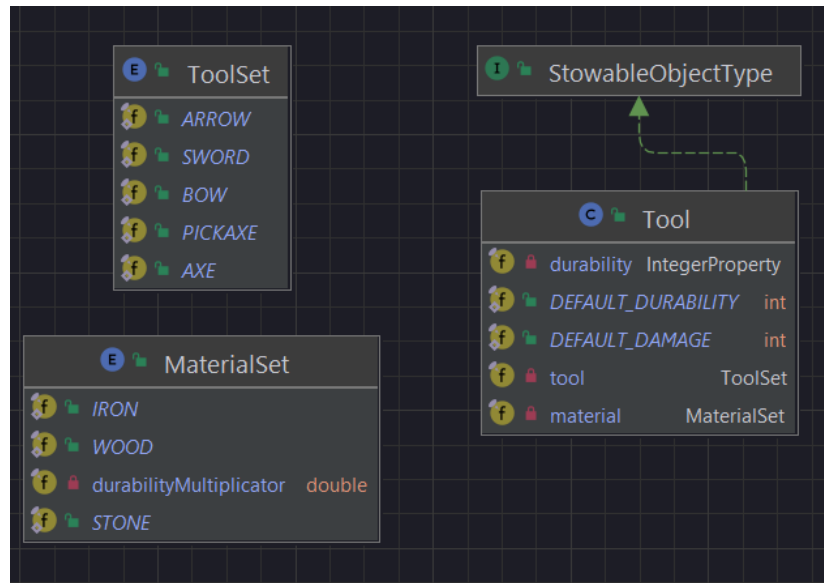


Les hautes herbes est une entités de type reproducteur et avec l'arbre, ils ont des entités qui peuvent être invocables.

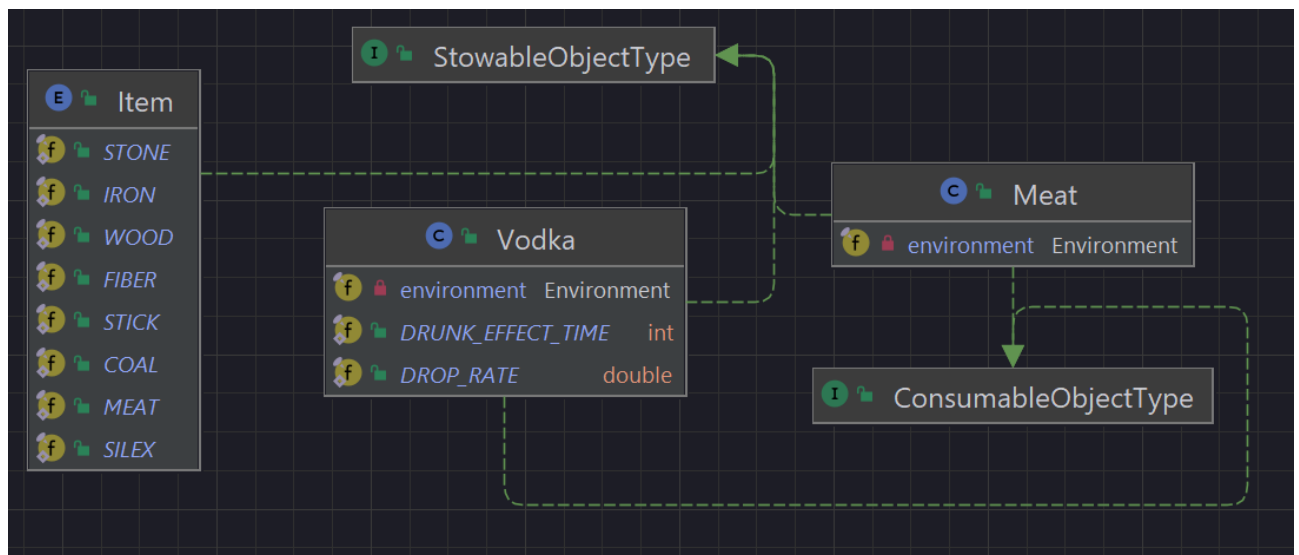


Ils ont bien entendu tous être placés sur la carte et tous être cassés. Ils sont stockables dans l'inventaire.

### Vue d'ensemble des classes gèrent les outils du jeu

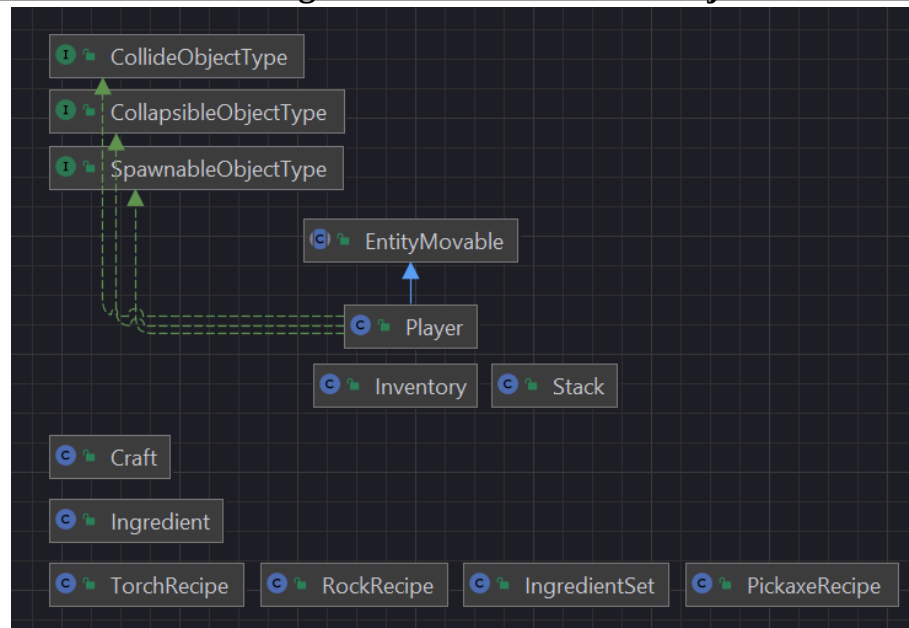


### Vue d'ensemble des classes gèrent les items du jeu



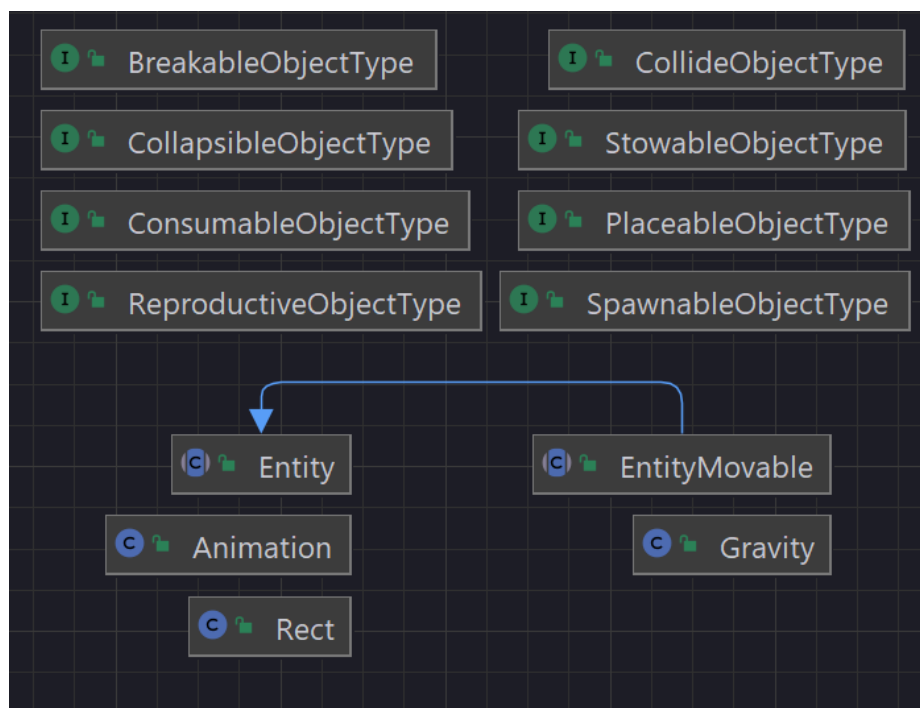
Le liquide transparent à effet secondaire est un consommable, de même pour la viande. Pour ces deux diagrammes, ils ont tous des enums qui permettent de visualiser tous les items et outils disponibles, qui ne possèdent pas de priorité particulière.

## Vue d'ensemble des classes gèrent les fonctionnalité du joueur dans le jeu



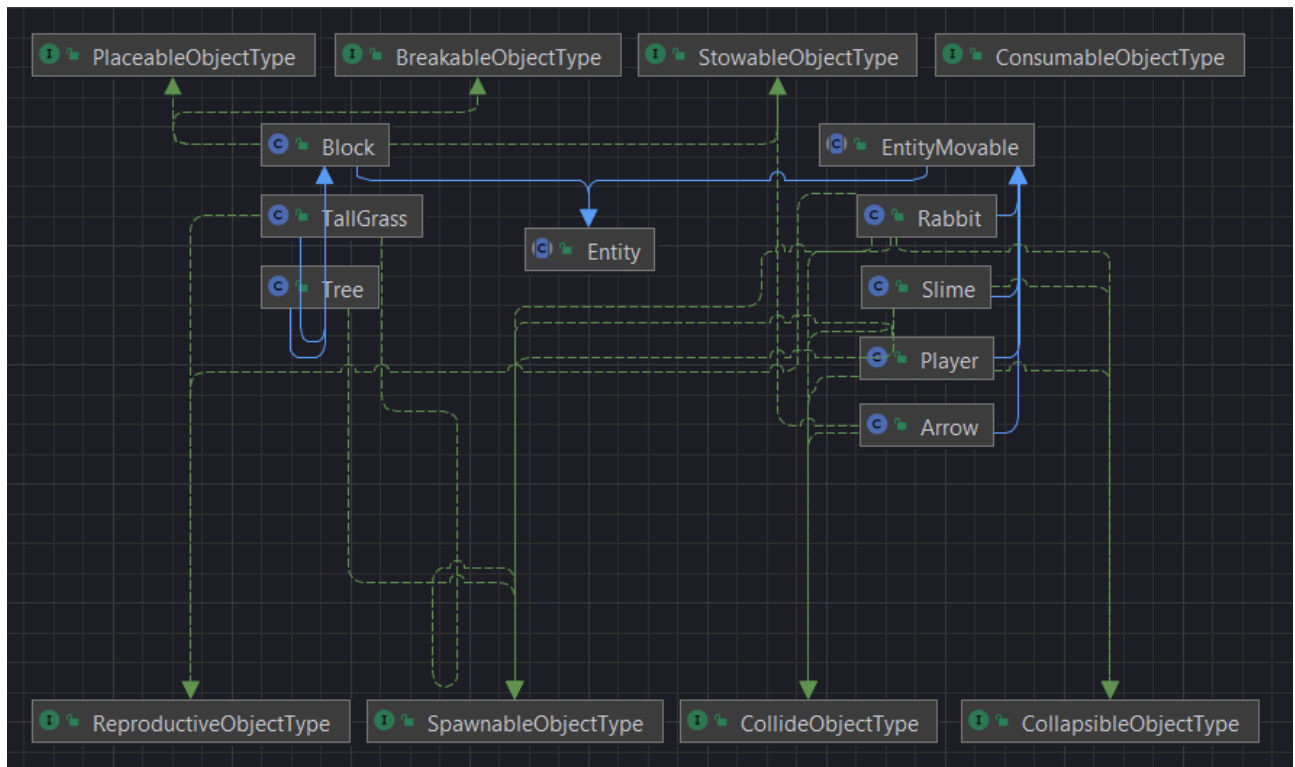
Le joueur hérite de EntityMovable pour pouvoir avec les méthodes qui permet de le faire déplacer.  
Le joueur possède en 1..1 la classe Inventory, Inventory à 0..\* stacks  
Le joueur connaît les recettes des constructions, c'est pour cela que je l'ai mis dans le diagramme.

## Vue d'ensemble des classes en relations avec les entity



Ce diagramme visualise tout ce qui concerne les choses qui vont avec les entités et les fonctionnalités liées.

## Vue d'ensemble sur les implémentations et l'héritage des entités



Je dirait que ce diagramme est plutôt intéressant sur le point de vue de la programmation objets. Les nombreux implémentations désignent une fonctionnalité différente :

- CollideObjectType : Les entités qui doivent entrer en collision avec son environnement.
- CollapsibleObjectType : Les entités qui doivent subir des dégâts, qui peuvent être touchées pour produire une action.
- SpawnableObjectType : Les entités qui peuvent être invoquées par quelque chose ou quelqu'un.
- ReproductiveObjectType : Les entités qui peuvent se reproduire.
- PlaceableObjectType / BreakableObjectType : Les entités qui peuvent être placées ou bien cassées.
- StorableObjectType : Les entités qui peuvent être rangées dans un inventaire qui contiendra des stacks de StorableObjectType.

EntityMovable : Entités qui sont bougeables.

Entity : Les entités qui ont des coordonnées en x et en y et qui sont plutôt statiques.

Pourquoi celui-là et pas un autre ? Car c'est la partie où il demande le plus d'héritage.