

- **New trait InteractivSim.** Goal: Give possibility to interact with other Sim
 - Keep track of the state of each Sim it can interact with ()
 - Interact = apply a function to a Sim
 - Usefull for example if you want to model influence of weather on cereals, during the time their growth
- **New abstract class “SimCommodity”.** Goal: simulate some commodities in order to have a more complex simulation
 - It extends Commodity (saleable, purchaseable, consumable)
 - It extends InteractivSim
 - e.g simulate each cereal individually, to measure its dryness, its weight, if insects are eating it ...
- **class ProductionLineSpec:**
 - New class parameters: “simulated_commodities: List[Commodity]”, and a function: (List[Commodity]) => (Commodity, Int)
 - Type not SimCommodities, in case you want basic plastic bag factory, without simulating the bags, the produced function does not depend on state of simulated_commodities (always the same, no outside influence)
 - produced is not known in advance, but can be computed with the function
 - Goal is to have a production that can change over time, depending on the state of the simulated_commodities, and the function
 - For example, adding fertilizer change the state of cereals (the simulated_commodities). They are heavier -> bigger produced
- **class Factory** now extend InteractivSim. It can influence SimCommodities of each productionLineSpec
 - Now contain a attribut “simulatedCommodities : List[SimCommodity]”
 - It should have a list of productionLineSpec, in order to induce more independencies
 - ex: You have 2 crops, and decide to use fertilizer only on 1, thus you impact only one productionLineSpec
 - add optional requires (Prod Line can run without them, but can be used in intermediate steps)
 - on the algo function of the production line, there should be something listening/awaiting on owner/other events:
 - * Listening: e.g in a farm factory. The use of fertilizer is optional, i.e your cereals can still growth without it.
 - * Awaiting: You need to harvest before being able to get your “produced” cereals in the end.
 - * ex: in a wheat production line, an event could be use fertilizer.
 - * another one could be external, and would be the weather.
 - * It should be used to influence the “frac” attributs, influencing the productivity

Comment inclure une list de plantes de blé dans la production line ? ou est ce

que ca doit être ajouter autre part ? -> peut-être directement dans la ferme, -> une methode `interactWithAgents` for each intermediate/optional step

```
class IntermediateStep { def InteractWithAgents(List[Sim]) { //pesticide ->
n_insects on wheat - //fertilizant -> cereals grows ++ } }
```

production line have a list of Actions each Action: - have some requires - have some effects on productivity and/or other sim objects

```
Action(requires:List[Commodities], effect: (List[Agents]) => (return nothing))
```

!!!!!! How to models each cereals inside the Farm ? -> new type of Sim, ones that are linked with some others: e.g a silos is linked with each cereals that are simulated inside it. Farm extends SimO: - inventaire {monai, tracteur, champ de blé, champ de maïs} - List[factory] (make sens to have sim0 inside sim0), represent multiple crops,...