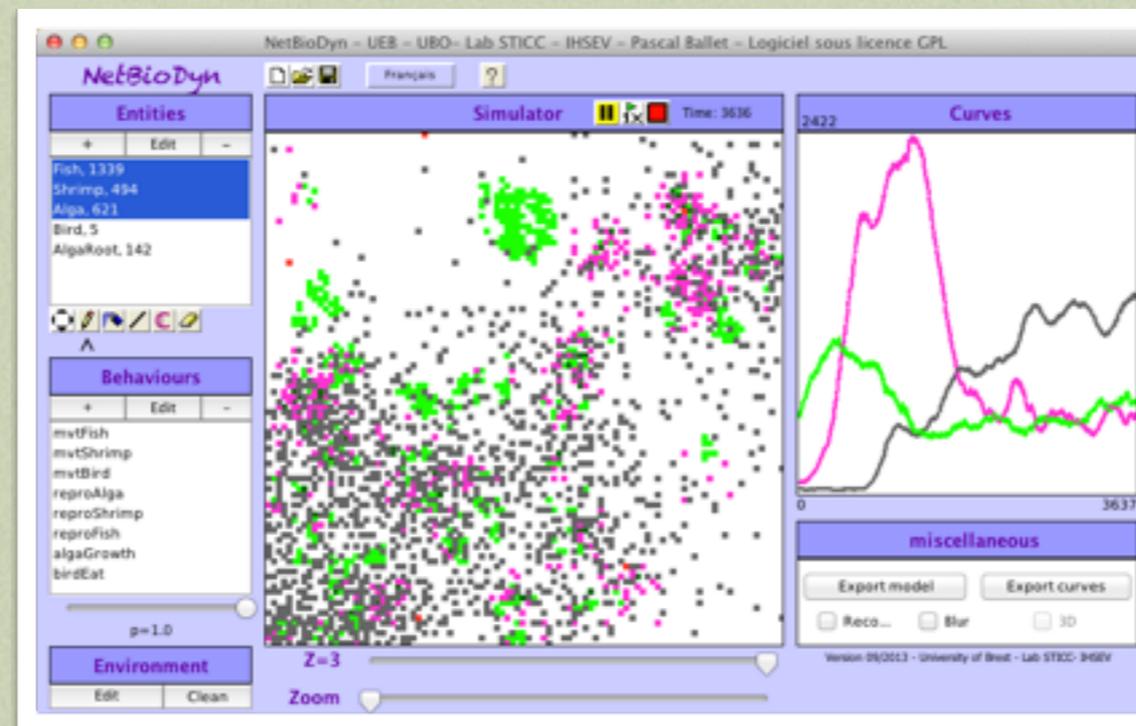


Pascal Ballet

# EXTENSIVE FISH FARM SIMULATION



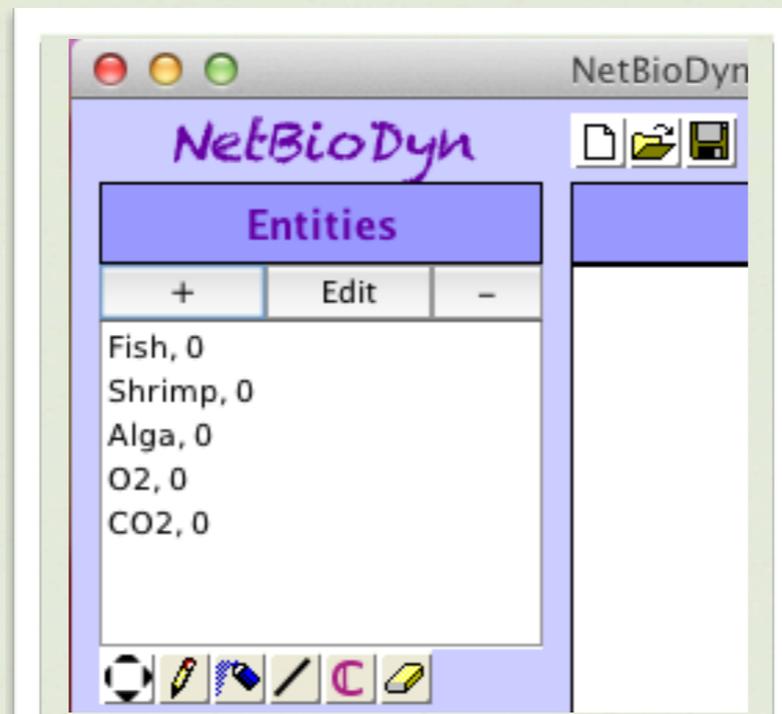
FOODSIM 2014 - TUTORIAL

# EXTENSIVE FISH FARM SIMULATION

An aquaculture farm located in Spain produces 1,500 tons of sea fishes and shrimps per year.

The farm is based on an auto-regulated ecosystem implying alga, shrimps, fishes and birds.

The aim of this tutorial is to simulate this farm by using an individual based model with NetBioDyn.



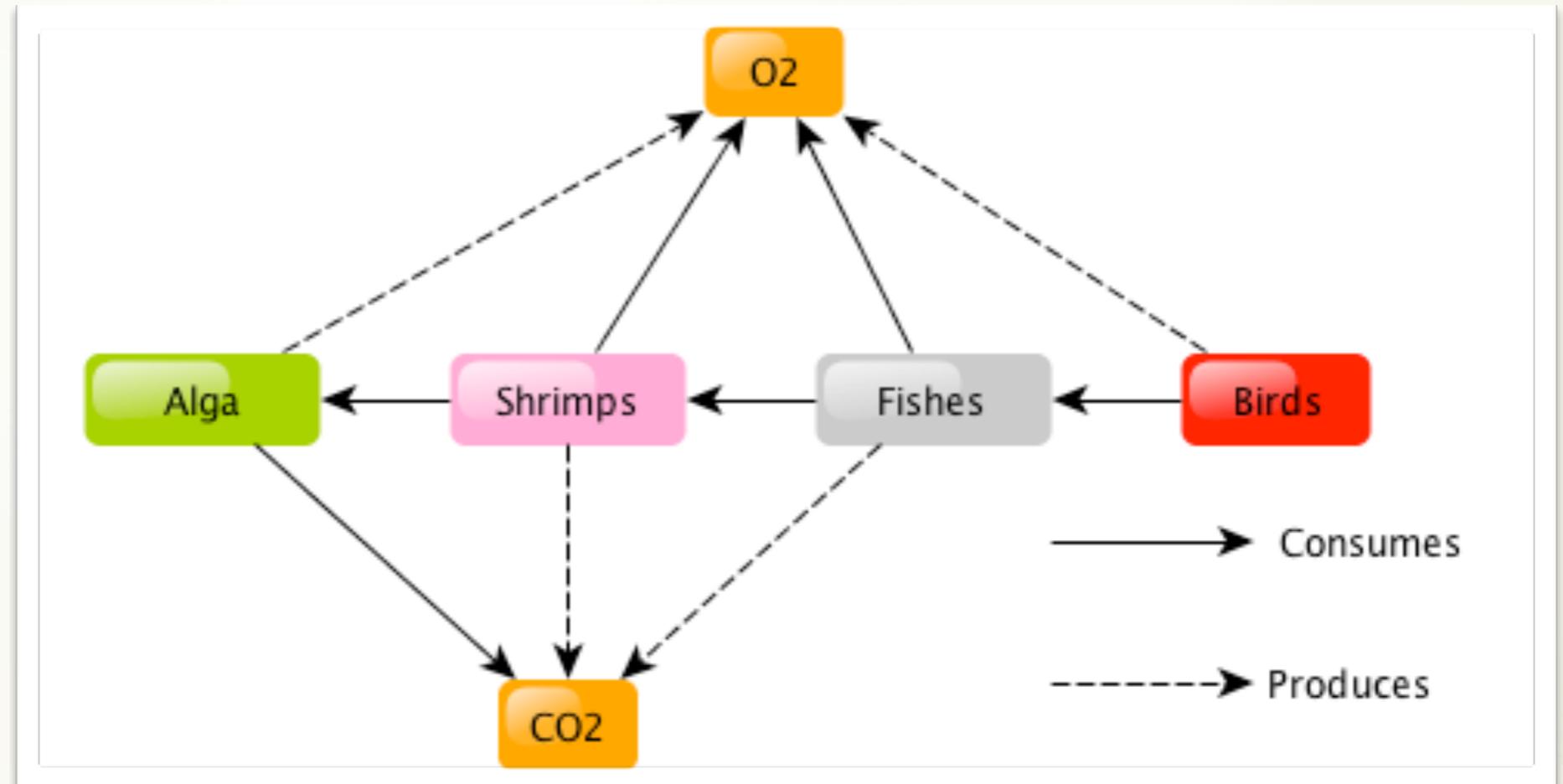
## Data

1. 4 kg of fish per  $m^3$ .
2. One fish weights 1kg.
3. Birds eat 20% of the fishes.
4. One pool is  $1km^2$ .
5. One simulation step equals 1 hour in the real world.
6. The fish production, in one year is 37 tons.
7. Only the fishes that have at least 3 years can be fished. A fish can live up to 30 years.

• Section 1 •

### Ecosystem

1. Alga. They produce  $O_2$ , consume  $CO_2$  and then can multiply.
2. Shrimps. They eat alga and can reproduce, consume  $O_2$  and produce  $CO_2$
3. Fishes. They eat shrimps, can reproduce, consume  $O_2$  and produce  $CO_2$
4. Birds. They eat fishes. Moreover, by stirring the water surface, they add  $O_2$ .
5.  $O_2$ . It moves at random in the water (Brownian motion).
6.  $CO_2$ . It moves at random in the water.



## Ecosystem Model

The ecosystem described in the above diagram takes place in a  $1 \text{ km}^2$  area of salt water (for about 2 meters deep). The diagram shows the dependencies between entities. These relations induce an self-regulated ecosystem.

The aim of the simulation is to explore the impact of change in this system in order to:

- 1-Evaluate the complexity of the system behaviour
- 2-Test hypothesis on the system for example by testing unexpected situations like disease or over-fishing.

• Section 2 •

## Entities

1. Alga and AlgaRoot. Size = 1x1x1, half-life = 100, color = green.



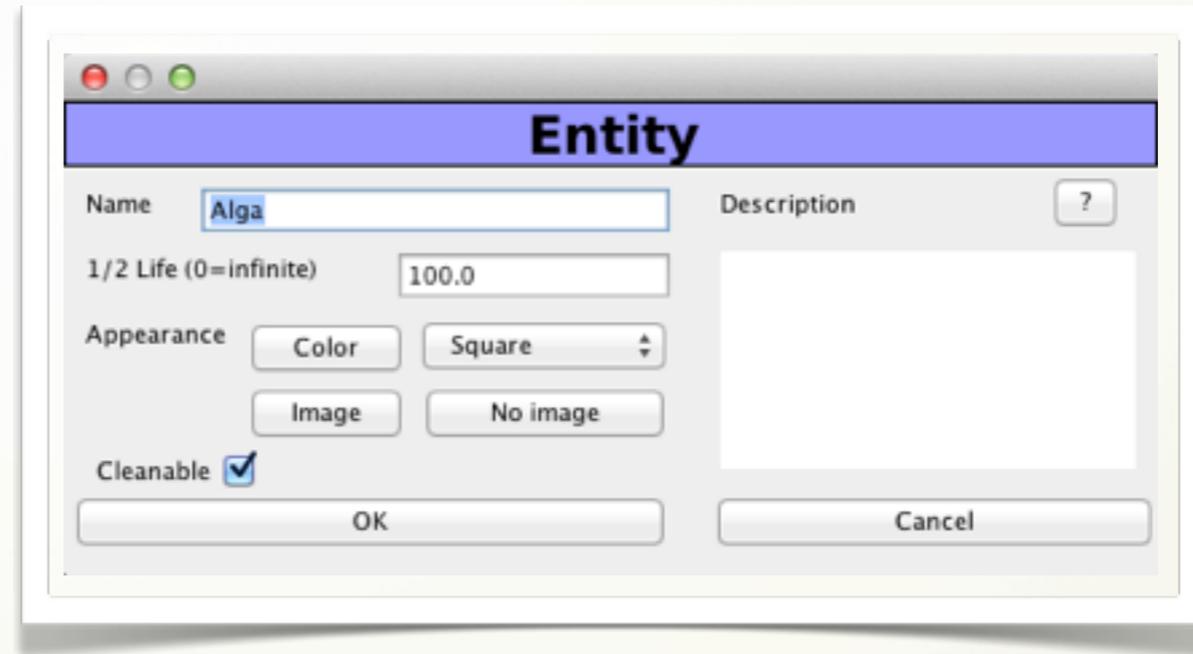
2. Shrimps. Size = 1x1x1, half-life = 100, color = pink.



3. Fishes. Size = 1x1x1, half-life = 200, color = grey.



4. Birds. Size = 1x1x1, half-life = infinity, color = red.



## Entities

Each entity is simply defined by its size (always 1x1x1 in our example), its half-life (time required to have 50% of a population to be destroyed: at each time step, the probability for an entity to die is  $p = 1 - e^{-dt/half\ life}$ ) and its color.

1 entity can represent more than 1 real entity.

Moreover, the position of each entity is stored in the environment matrix.

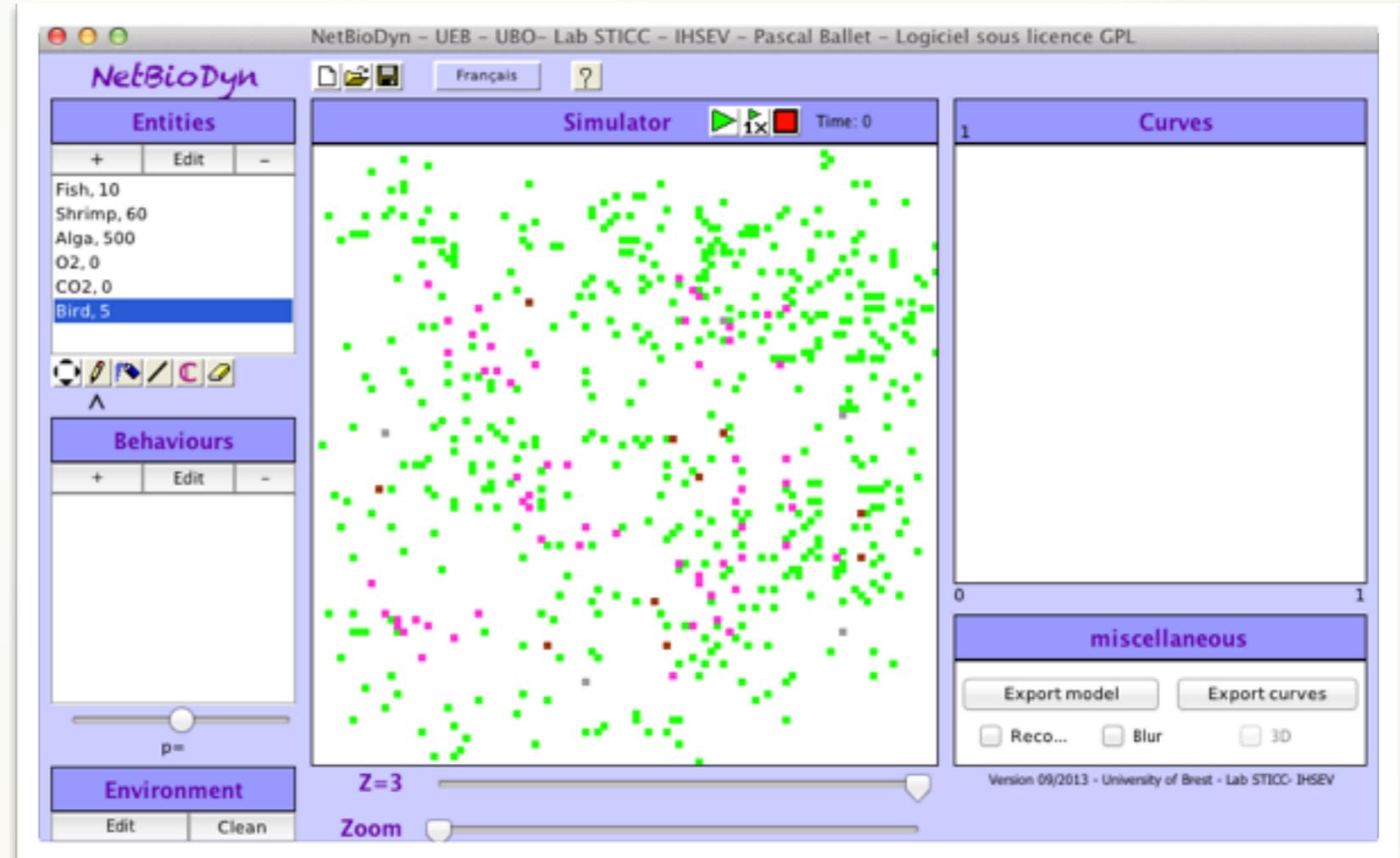
Note that two entities can not be at the same location at the same time.

In order to simplify this example, we do not take into account the molecules O<sub>2</sub> and CO<sub>2</sub>.

## • Section 3 •

### Description

1. The simulation environment is a (100x100x4) matrix.
2. The first layer of the matrix ( $z=0$ ) is the place of the alga (in green).
3. The second layer ( $z=1$ ) is where the shrimps are located (in pink).
4. The third layer ( $z=2$ ) is the layer of fishes.
5. The fourth layer ( $z=3$ ) is for the birds.
6. The molecules  $O_2$  and  $CO_2$  move all accross the matrix.



## Initial state

The matrix 100x100x4 contains all the entities.

At  $z=0$ , 500 alga are placed at random.

At  $z=1$ , 60 shrimps are placed at random.

At  $z=2$ , 10 fishes are placed at random.

At  $z=3$ , 5 birds are placed at random.

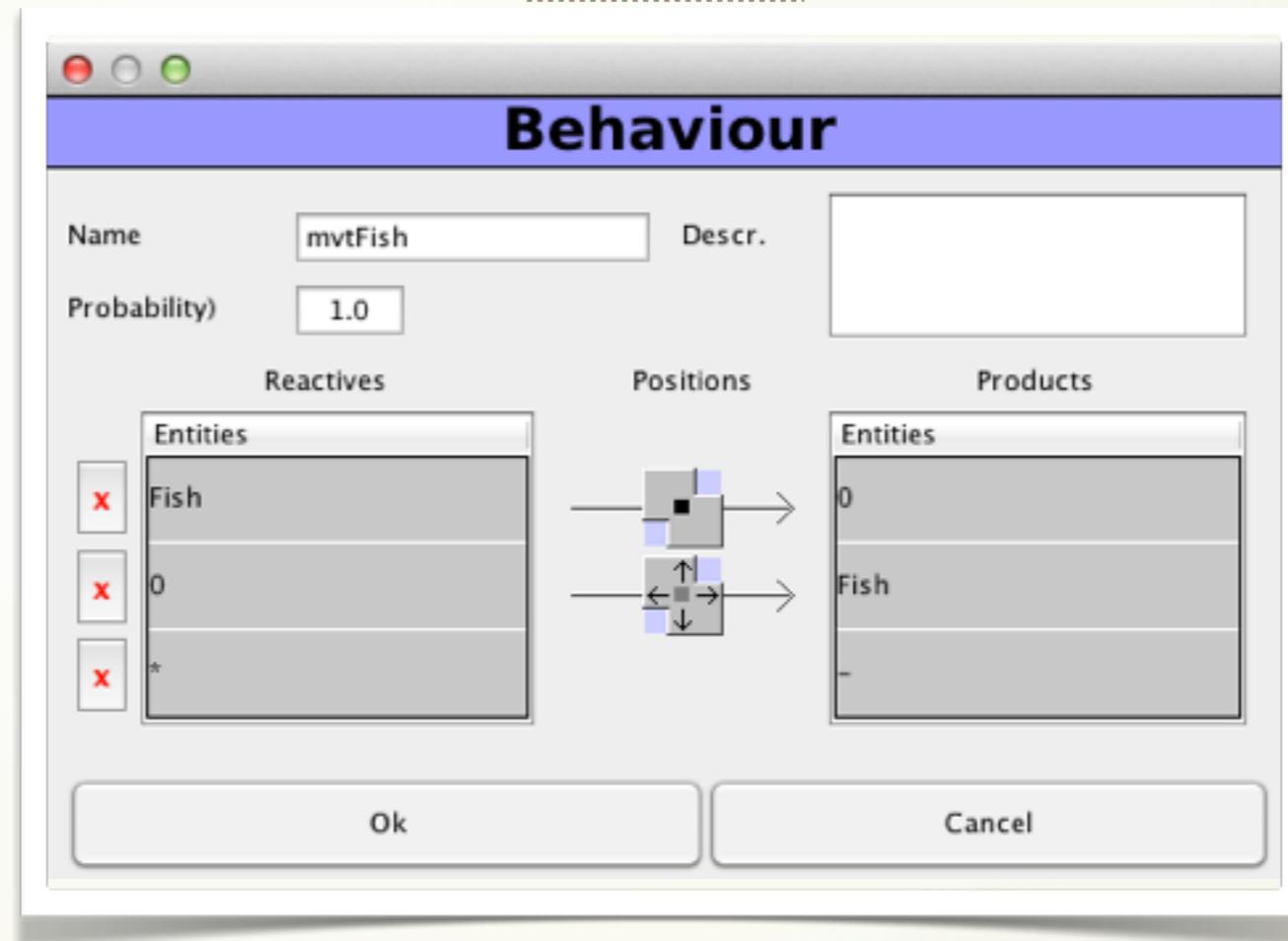
The screen capture of NetBioDyn shows the entities at the top-left corner and their positions are the screen center.

The layer change is made by the slider Z at the bottom of the user interface.

• Section 4 •

Description

1. mvtFish: movement of fishes.
2. mvtShrimp: movement of shrimps
3. mvtBirds: movement of birds
4. reproAlga: proliferation of alg.
5. reproShrimp: reproduction of shrimps if alga is eaten.
6. reproFish: reproduction of fishes if shrimp is eaten.
7. birdEat: bird eat fishes.
8. algaGrowth: growth of alga from root to eatable alga



## Behaviours

A behaviour is defined for a maximum of three reactives and three products.

The behaviour is located in the matrix for each reactive and product:

- the first reactive is at the center of the behaviour (but anywhere in the matrix)
- the second and third reactive are located around the first reactive.

- By selecting the arrows in the possible positions, the user indicates where the second and the first reactives can be according to the first reactive (left, right, up, down, bottom or up).
- The products give the transformation of the reactive.
- The probability is tested when reactives are all in situation

- Section 5 •

# Behaviours details

Name:  Descr.:

Probability:

Reactives	Positions	Products
Entities: Shrimp		Entities: 0
0		Shrimp
*		-

Name:  Descr.:

Probability:

Reactives	Positions	Products
Entities: Bird		Entities: 0
0		Bird
*		-

Name:  Descr.:

Probability:

Reactives	Positions	Products
Entities: Alga		Entities: Alga
0		Alga
*		-

Name:  Descr.:

Probability:

Reactives	Positions	Products
Entities: Shrimp		Entities: Shrimp
0		Shrimp
Alga		AlgaRoot

Name:  Descr.:

Probability:

Reactives	Positions	Products
Entities: Fish		Entities: Fish
0		Fish
Shrimp		0

Name:  Descr.:

Probability:

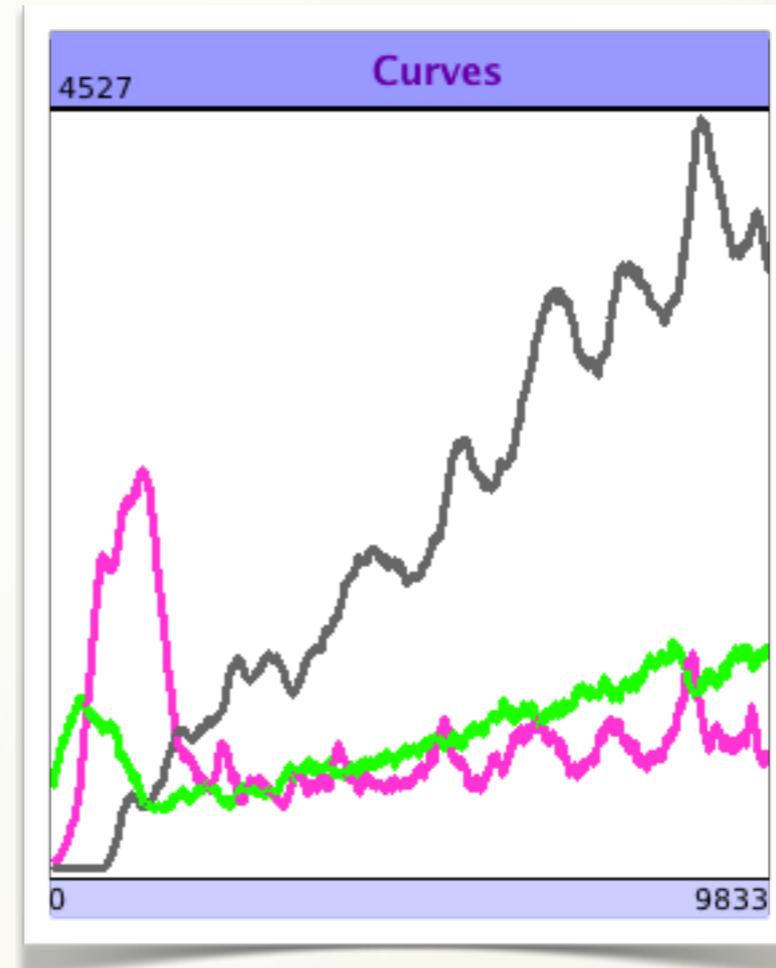
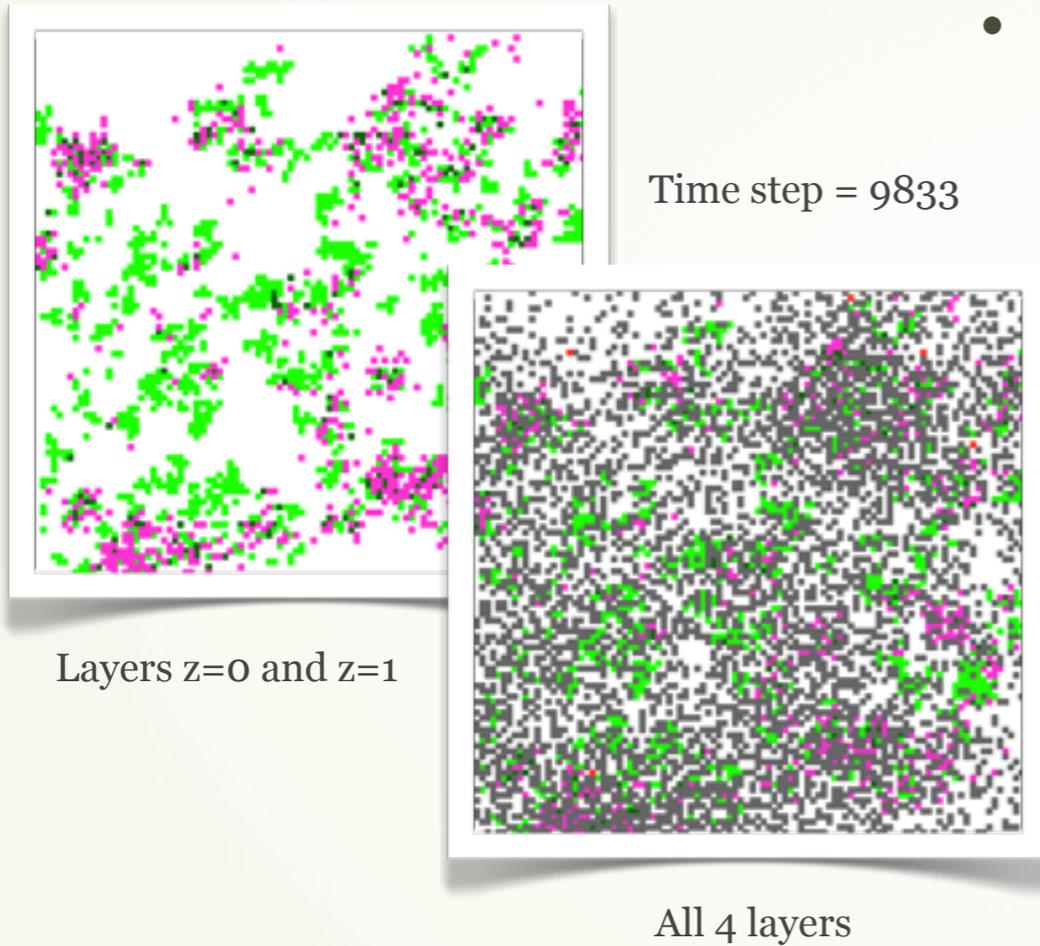
Reactives	Positions	Products
Entities: AlgaRoot		Entities: Alga
*		-
*		-

Name:  Descr.:

Probability:

Reactives	Positions	Products
Entities: Bird		Entities: Bird
Fish		0
*		-

• Section 6 •



## Simulation

When all entities and all behaviours are defined, after placing the entities in the environment matrix, the simulation can be executed.

The results are shown in the matrix and in the curve panel. In the matrix, the spatial location of entities is shown at real time and in the curve panel, the population of selected entities are displayed and compared.

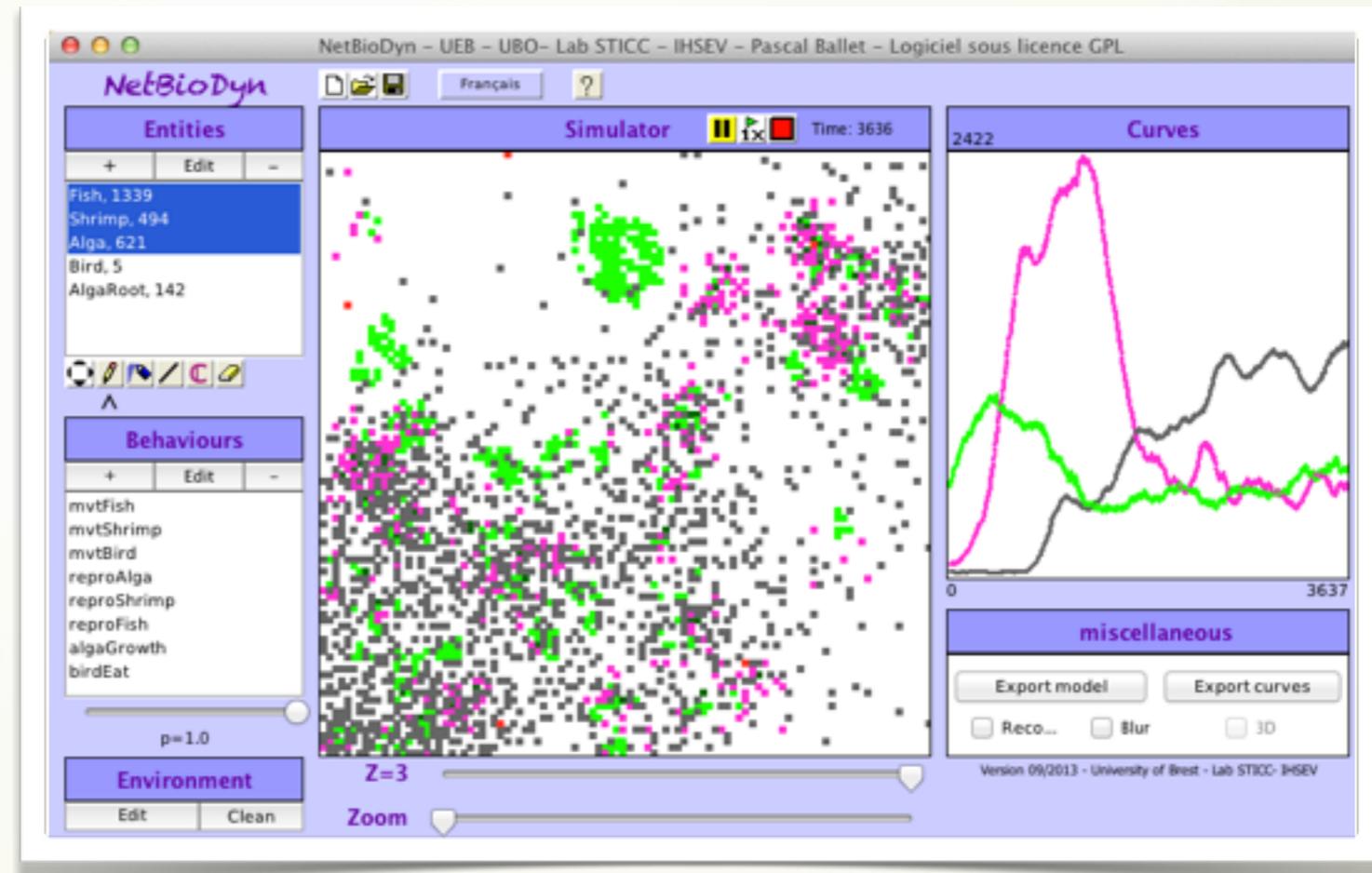
Note that it is always possible to change an entity, a behaviour or the initial state at any time in the process of modelling.

The curves can be exported in a text file to be included into software like Excel for more diversified treatments and analysis.

## • Section 7 •

### Pro - Cons

- + entites and behaviours are simple to define
- + the initial state is drawn like in a paint software
- + a result is quickly given
- + the addition - deletion - modification of a model is easy
- + interesting for teaching
- the model parameters can be hard to tune
- the software is monoscale
- only one simulation is possible at once
- the simulation time can be long



## Conclusion

The modelling of ecosystems with NetBioDyn allows non-specialists to create complex simulations without coding.

The dynamic of eco-systems can be experimented by students in a simple way.

However, when the number of entities and behaviours become big (more than 20), the comprehension and the control of the model becomes difficult.