Kelp forests

Lesson about the effect of sea otters on global warming

DynaLearn level 3 | Version 1.0

Summary	
Kelp forests are important for absorbing carbon dioxide and reducing the greenhouse effect. This lesson emphasizes the importance of promoting kelp forests. Sea urchins can destroy these kelp forests, but the presence of sea otters can reduce the number of sea urchins by eating them. Unfortunately, sea otters are almost extinct due to hunting. There should be a ban on hunting sea otters and breeding programs could be established to reintroduce them. The presence of sea otters reduces the foraging behaviour of sea urchins, causing them to eat less kelp and spend more energy on bodybuilding, which improves their health.	
Given name	
Surname	
Class	
Date	
Comments by teacher	

1. Starting DynaLearn

There are several ways to log in. Use one of the two options below. Then check whether the login was successful (see 'Let's check').

Via a code:

1. Go to DynaLearn (https://create.dynalearn.nl/).
2. Click on 'log in with code', at the bottom left.
3. Enter the project code and your (school) email address.
4. Copy the code from the confirmation email received from *dynalearn.nl* (see spam folder if needed) and fill in the other details.
5. Log in to DynaLearn.

By email invitation:

1. Copy the login details from the invitation email received from *dynalearn.nl*.
2. Go to DynaLearn (https://create.dynalearn.nl/).
3. Log in to DynaLearn.

Let's check!

After logging in, you will automatically enter in the white workspace of the assignment. You can recognize it by the question mark on the right side of the screen  . Is the question mark missing? Then first do the following:

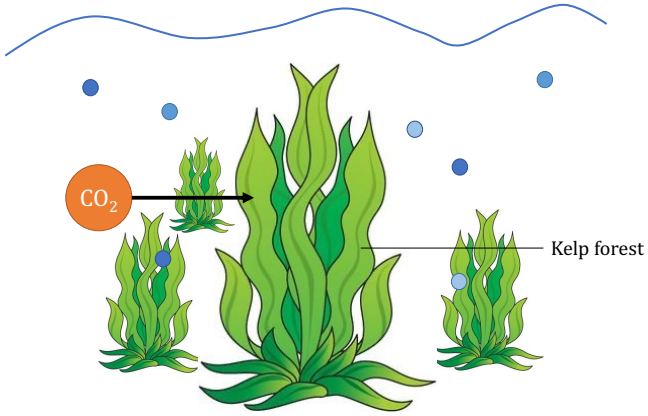
- In DynaLearn, click  . Click on 'Select norm model'.
- Choose 'Kelp forests' and press 'Load'.

Save model file and start:

1. Click on  top left. Change the name to 'Kelp forests' and click 'Save'.
2. How do you proceed? Just follow the steps in this workbook. Note! You can't skip steps. Ask for help if you can't figure it out at a certain step. The video function  in DynaLearn shows how a model ingredient can be made. The sources contain information about kelp forests and the boxes contain a brief explanation about the model ingredient. Put a check mark √ next to step you performed. This way you keep track of where you left off.

2. Kelp forests and carbon dioxide

1. Read Source 1.



Source 1. Kelp can grow into large kelp forests off the coast. Kelp absorbs carbon dioxide for photosynthesis and thus contribute to a reduction in global warming.

2. Read Box 1.

Box 1. Entity and quantity.

An entity is usually a physical thing in a system (e.g., car, human).

A quantity is a measurable property of an entity (e.g., temperature, length).

3. Create the **entity** *Kelp* (see →).

4. Create the **quantity** *Amount* of the entity *Kelp* (see →).

5. Read Box 2.

Box 2. Help function.

If the question mark or an ingredient in your model is red, then something is wrong. Click the question mark for a hint. Then click on a number, for example 1 to see where the error is in your model. Only use the question mark if you can't figure it out yourself!

6. Create the entity CO_2.

7. Create the quantity *Amount* of the entity CO_2.

8. Read Box 3.

Box 3. Cause-and-effect relationships.

In DynaLearn, there are two types of cause-and-effect relationships:

- Positive relationship : the quantities change in the same direction (if quantity 1 increases, then quantity 2 increases also)
- Negative relationship : the quantities change in opposite direction (if quantity 1 increases, then quantity 2 decreases. Or vice versa: if quantity 1 decreases, then quantity 2 increases)

9. Make the cause-effect relationship (⊕ or ⊖) between the quantity *Amount* of the entity *Kelp* and the quantity *Amount* of the entity CO_2 (see →).

10. You can keep the model organized and clear by using a few buttons at the bottom of the screen. Click to align everything neatly. Click to make your model fit on the screen. Use these buttons regularly.

11. Read Box 4.

Box 4. Change of a quantity.

A quantity can change. This is indicated by . The delta symbol (δ) is the mathematical sign for change (also called the derivative). The down arrow (▼) is a decrease, the zero (Ø) is constant and the up arrow (▲) is an increase.

12. You can now run a simulation. Therefore, you must first indicate an initial change. Start with an increase for the amount of kelp (see →).

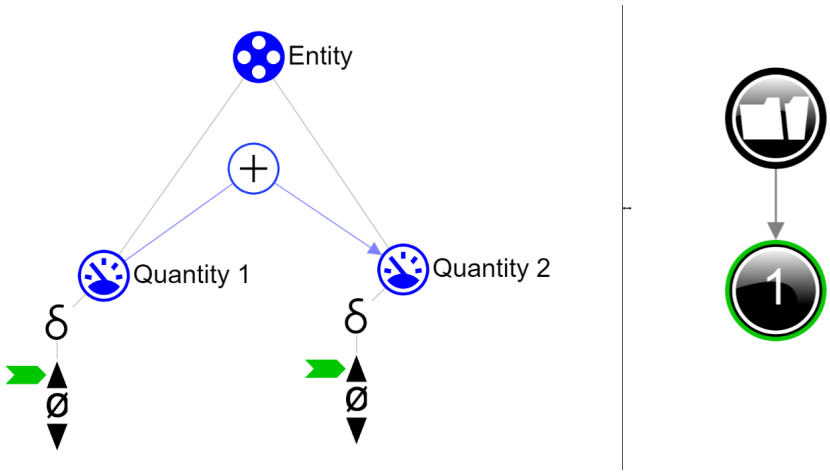
    a. Click at *Amount* of the entity *Kelp* on ▲ and choose . It should look like this:

13. Simulate your model by clicking on .

14. Read Box 5.

Box 5. Read the outcome of a simulation.

After starting a simulation (with), a window appears on the right in which the possible states of the system are indicated. There is one possible state in this example .



You can click on the state to view the outcome. The state icon then gets a green circle. In the model, the change for this state is indicated by green arrows . The model shows that in state 1 *Quantity 1* increases and that therefore *Quantity 2* also increases.

15. Return the result of the first simulation in the table below (cross out incorrect answers). Then start two simulations in which the amount of kelp **remains the same** and **decreases.** Return these results in the table.

Amount of kelp...	**Amount of CO_2...**
Increases	*Decreases/stays the same/increases*
Stays the same	*Decreases/stays the same/increases*
Declining	*Decreases/stays the same/increases*

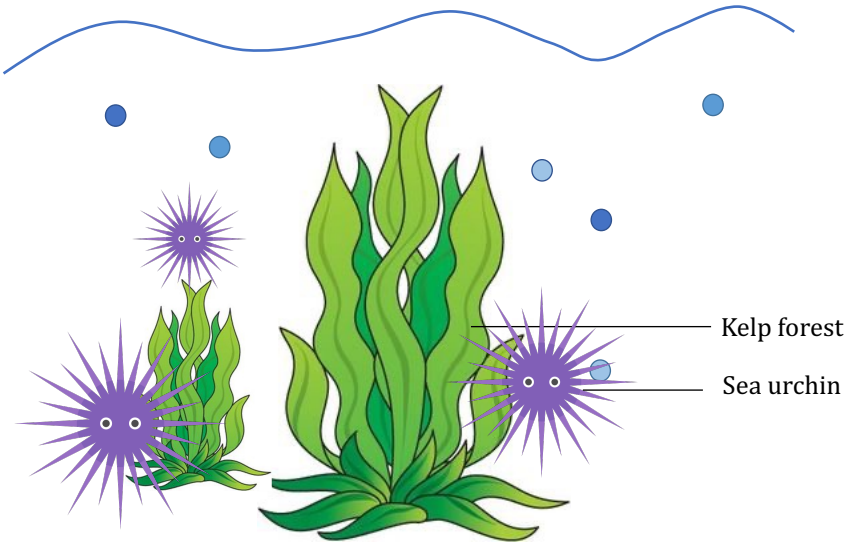
16. Read Box 6.

Box 6. Help function.

If the explanation mark appears (!), something is wrong during the simulation. Click on the explanation mark (!) for a hint. Then click on a number, for example (1), to see where the error is in your simulation.

3. Sea urchins and Kelp

1. Read Source 2.



Source 2. Sea urchins are real gluttons. They eat away so many kelps in a short time that the forests can no longer recover and are largely lost.

2. Create the entity *Sea urchin*.

3. Create the quantity *Number* for the entity *Sea urchin*.

4. Make the relationship (⊕ or ⊖) between the quantities *Number* of the entity *Sea urchin* and *Amount* of the entity *Kelp*.

5. Click to align everything neatly. Click to make your model fit on your screen.

6. Set as initial change (see →):
   a. Remove the initial change at *Amount* of the entity *Kelp* (click and).
   b. Set *Amount* of the entity *Sea urchin* to increase.

7. Simulate your model, look at the results and cross out the wrong answer.

If the number of sea urchins increases, then the amount of kelp will *increase/remain the same/decrease.* As a result, the amount of carbon dioxide will *increase/remain the same/decrease.*

4. Sea otters eat sea urchins

1. Read Source 3.

Source 3. Researchers have found that the presence of enough sea otters has a great effect on the presence of sea urchins. Despite the sharp spines, sea otters eat a lot of sea urchins, they are their natural enemies. However, sea otters have been nearly wiped out by fur hunting for their fur. Therefore, there must be a ban on hunting sea otters. In addition, breeding programs can also be set up to release young sea otters into the area.

2. Create the entity *Sea otter*.

3. Create the quantity *Number* for the entity *Sea otter*.

4. Make the relationship (⊕ or ⊖) between the quantities *Number* of the entity *Sea otter* and *Number* of the entity *Sea urchin*.

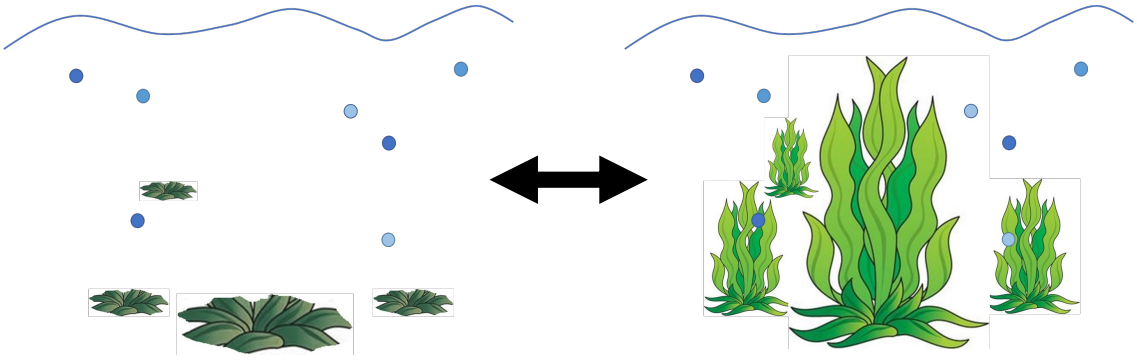
5. Set as initial change (see →):
   a. Remove the initial change at *Number* of the entity *Sea urchin* (click and).
   b. Set *Number* of the entity *Sea otter* to increase.

6. Simulate your model, look at the results and cross out the wrong answer.

If the number of sea otters increases, the number of sea urchins will *increase/remain the same/decrease.* As a result, the amount of kelp will *increase/remain the same/decrease.*

5. Kelp: from wasteland to forest

1. Read Source 4.



Source 4. If there are many sea urchins, they eat all the kelp and then a wasteland emerges. If there are enough sea otters, there are fewer sea urchins and then the kelp can grow into a forest.

2. Read Box 8.

Box 8. Quantity space.

A newly created quantity does not yet have a quantity space. By adding a quantity space, you can indicate which values a quantity can take on. A quantity space consists of alternating points (●) and intervals (▮).

- A *point value* is only one value. For example, a *boiling point*. A special point is the zero point, for which there is a separate symbol (Ø) in DynaLearn.
- An *interval* is a set of values. The liquid phase of a substance is an example of an interval. In the case of water, the interval 'liquid' contains all values between 0 °C and 100 °C. The values 0 °C and 100 °C are respectively the 'melting *point*' and the 'boiling *point*' between which the interval 'liquid' is located.

3. Create a quantity space (see →) for the quantity *Kelp* with a *Transition point*, above it an interval (*Forest*) and below that an interval (*Wasteland*). It will look like this:

4. Set up initial value (see →):
   a. Set the initial value of the quantity *Amount* of the entity *Kelp* to *Wasteland* (click *Wasteland* and choose).
   b. Let *Number* of the entity *Sea otter* stand on increase.

5. Simulate your model, look at the results and cross out the wrong answer.

How many consecutive states do you get?
1/2/3
What value does the quantity *Amount* of the entity *Kelp* achieve? *wasteland/transition point/forest*

6. Read Box 9.

Box 9. What is an exogenous influence.

If you want a quantity to decrease, remain stable or increase throughout the simulation, you must add an external influence (an exogenous influence) to the quantity.

7. Set up initial value:
   a. Create an increasing exogenous influence for *Number* of entity *Sea otter* (see →).
   b. Leave the initial value of the quantity *Amount* of the entity *Kelp* to *Wasteland*.

8. Simulate your model, look at the results and cross out the wrong answer.

How many consecutive states do you get?
1/2/3
What value does the quantity *Amount* of the entity *Kelp* achieve? *wasteland/transition point/forest*

6. Sea otters change the foraging behaviour of sea urchins

1. Read Source 5.

Source 5. Sea otters change the foraging behaviour of sea urchins. To avoid being eaten, they become passive and eat fewer kelp.

2. Create the quantity *Foraging behaviour* of the entity Sea *urchin*.

3. Create a quantity space (see →) for the quantity *Foraging behaviour* with a *Transition point*, above that an interval (*Active*) and below that an interval (*Passive*).

4. The number of sea otters therefore has an effect on foraging behaviour. Subsequently, the foraging behaviour effects the amount of kelp. Make these two relationships (or).

5. Read Box 10.

Box 10. Correspondence.

In a system, it can happen that certain values of different quantities can only occur together. You can then make a **correspondence** between the quantity spaces of those quantities.

6. Create a correspondence of type directed inversed (see →) between the quantity space of *Foraging behaviour* of the entity *Sea urchin* and the quantity space of *Amount* of the entity *Kelp*.

7. Set up initial value:
   a. Remove the initial value *Wasteland* (click and) from the quantity *Amount* of the entity *Kelp.*
   b. Set the initial value of the quantity *Foraging Behaviour* of the entity *Sea urchin* to *Active*.
   c. Leave the increasing exogenous influence (see →) for *Number* of the entity *Sea otter.*

8. Simulate your model, look at the results and cross out the wrong answer.

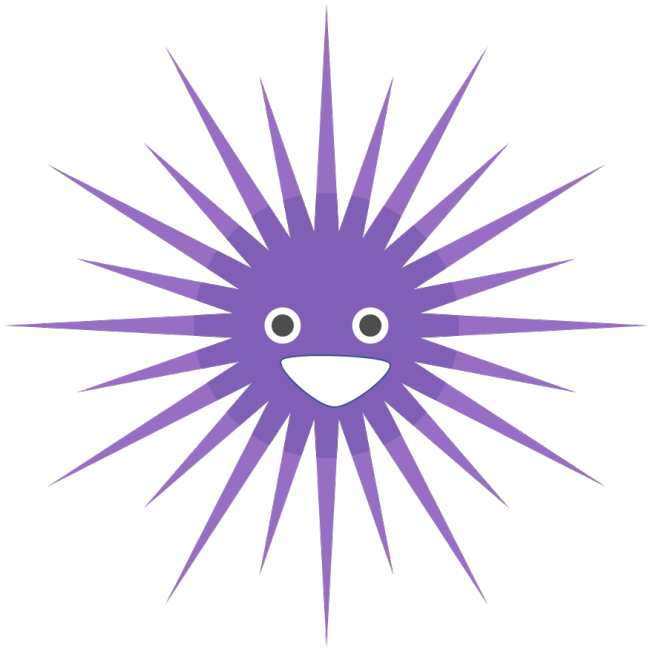
How many consecutive states do you get? *1/2/3*

9. View states 1 and 3 of the simulation result. Answer the question below (cross out the wrong answer).

State 1 The sea urchins are *passive/active* and the amount of kelp is *wasteland/forest*. As the number of sea otters increases, foraging behaviour will increase/remain the same/decrease and the amount of kelp will *increase/remain the same/decrease.* **State 3** The sea urchins are *passive/active* and the amount of kelp is *wasteland/forest*.

7. Sea otters make sea urchins healthier

1. Read Source 6.



Source 6. Due to the presence of sea otters, the sea urchins lose less energy to the search for food. Because they are now much more passive in their eating behaviour, all energy now goes to the construction of their body and thus their health improves.

2. Create the quantity *Health* of the entity *Sea urchin*.

3. Create a quantity space (see →) for the quantity *Health* with a *Transition point,* above it an interval (*Healthy*) and below that an interval (*Sick*).

4. The foraging behaviour has an effect on health. Make this relationship (or).

5. Create a correspondence of type direct inversed (see →) between the quantity space of *Foraging behaviour* and the quantity space of *Health* of the entity *Sea urchin.*

6. Read Box 11. Is your model completely finished?

Box 11. Progress bar.

At the bottom of the screen is the *progress bar* (see example below).

Entity says: 4/4/0, this means: 4 created, 4 needed, 0 error. For quantity it says : 5/17/1: this means: 5 made, 17 needed, 1 error. If all the numbers are green, that type is settled.

7. Set up initial value (see →):
   a. Leave the initial values as in the previous simulation.

8. Simulate your model and look at the states 1 and 3 of the simulation result. Answer the question below (cross out the wrong answer).

State 1

The sea urchins are passive and the health is *sick/healthy*. As the number of sea otters increases, foraging behaviour will increase/remain the same/decrease and health *will increase/remain the same/decrease.*

State 3

The sea urchins are *passive/active* and the health is *sick/healthy*.