Module 5B
“One Fish, Two Fish..”

Building an Ocean Food Chain / Web model

Let’s build a model of an ocean ecosystem. Open a new NetLogo model page.

Create the Ocean

Go to Code section and type the following:

to setup
clear-all
ask patches [  
set pcolor blue  
]
end

Go to interface window. Choose “add” and “button”. Click anywhere in the white, a button will appear and a button dialog window will open. Type “setup” in the space provided and click OK. Now the button will be labeled setup. Click the button and your ocean appears!

Add some plants (producers)

Now let’s add some plants to our ocean. Return to the code section and add the lines in bold below to the setup section.

to setup
clear-all
ask patches [  
set pcolor blue  
]  
create-turtles 200  
ask turtles [  
set shape "plant"  
set color green  
setxy random-xcor random-ycor  
]
end

Return to the Interface page and press the “Setup” button again. You should see something like the image to the right. If you press setup repeatedly, what happens?
Add some fish (primary consumers)

Now let’s add the first level of consumers. It is easy to create more turtles, but we need to
distinguish the “fish” turtles from the “plant” turtles. So we are going to name some breeds and
change part of the setup code.

```
breed [plants]
breed [redfish]

to setup
  ca
  ask patches [  
    set pcolor blue
  ]
  create-plants 200
  ask plants [  
    set shape "plant"
    set color green
    setxy random-xcor random-ycor
  ]
  create-redfish 100
  ask redfish [  
    set shape "fish"
    set color red
    setxy random-xcor random-ycor
  ]
end
```

Again, press the setup button several times to see the effect of the random position functions.
Now we’ll add a second section of code to make the fish move.

Below the setup code, add these lines:

```lisp
to go
    ask redfish [
        fd 1
        eat
    ]
end

to eat
    if any? plants-here [
        ask one-of plants-here [die]
    ]
end
```

Then return to the interface page and add a “go” button. Be sure to check the forever box. Test your model.

**Agent interaction**

To model a food web, there needs to be agent interaction. So let’s add some code to have the fish eat the plants. We’ll add a new code section called “eat” and activate it from the go section.

```lisp
to go
    ask redfish [
        fd 1
        eat
    ]
end

to eat
    if any? plants-here [
        ask one-of plants-here [die]
    ]
end
```

The eat code checks to see if a fish and plant occupy the same patch, and then allows the fish to only eat one plant. Check and see what happens now.

We need to have the plants grow back. So add the following:

```lisp
to go
    ask redfish [
        fd 1
        eat
    ]
    grow-plants
go
end

to eat
    if any? plants-here [
        ask one-of plants-here [die]
    ]
end
```

to grow-plants
  ask plants [ 
    if random 100 < 8 [ 
      hatch 1 
      setxy random-xcor random-ycor 
    ] ] 
end

Run the model several times. You should notice that sometimes the plants disappear entirely and sometimes they overgrow the world. Why? What factors control populations in the real world?

Let’s change the code so that plants can only grow on empty patches:

to grow-plants
  ask patches [if not any? plants-here [ 
    if random 100 < 8 [ 
      sprout-plants 1 [ 
        set shape "plant"
        set color green 
      ] ] ] 
end

Run the model a few times and see what happens.

**Plots and monitors - outputs**

Now let’s create a monitor and a plot. From the Interface tab, select a monitor and place it in an open space. It will automatically open to the editing window. Name it “plants” and type “count plants” in the Reporter space. It should look like this:

![Monitor screenshot](image)

Now choose plot and place it an open space. Again it will open the editing window. Name the plot “Populations”. Change the first pen name to plants, the color to green and the pen update command to “plot count plants”. Add a second pen for the fish. It should look like this:
If you run the model now, you will note that the monitor is counting, but nothing is appearing on the plot. We need to add two lines of code for the plot to be updated.

At the end of the setup procedure, add the line “reset-ticks” and at the end of the go procedure, add the line “tick”.

```plaintext
to setup
  clear-all
  ask patches [ 
    set pcolor blue
  ]
  create-plants 200
  ask plants [ 
    set shape "plant"
    set color green
    setxy random-xcor random-ycor
  ]
  create-redfish 100
  ask redfish [ 
    set shape "fish"
    set color red
    setxy random-xcor random-ycor
  ]
  reset-ticks
end
to go
  ask redfish [ 
    fd 1
    set energy energy - 1
    eat
  ]
  tick
  grow-plants
  tick
end
```
Now run the model. What is the carry-capacity of your ecosystem? What variable(s) in the code results in this value? Experiment and discuss.

**Fish that reproduce and die**

Let’s add some code to have the fish reproduce and die at a certain age. We give the fish attributes of age and energy, set some values when first setting up the world, then limits on the values they can reach before the fish die. The fish gain energy from eating the plants, loose energy when moving and reproducing. Finally, we add a section for reproduction.

```plaintext
turtles-own [energy age]

to setup
  clear-all
  ask patches [  
    set pcolor blue
  ]
  create-plants 200
  ask plants [  
    set shape "plant"
    set color green
    setxy random-xcor random-ycor
  ]
  create-redfish 100
  ask redfish [  
    set shape "fish"
    set color red
    setxy random-xcor random-ycor
    set energy random 50
    set age random 20
  ]
  reset-ticks
end

to go
  ask redfish [  
    fd 1
    set energy energy - 2
    eat
    reproduce
    set age age + 1
    if age > 20 [die]
    if energy < 1 [die]
  ]
  grow-plants
  tick
end

to eat
  if any? plants-here [  
    ask one-of plants-here [  
      die
    ]
  ]
```

CAST---Module 5B: Building an Ocean Food Chain © Pittsburgh Supercomputing Center
set energy energy + 5
] end

to grow-plants
ask patches [if not any? plants-here [
if random 100 < 8 [
  sprout-plants 1 [
    set shape "plant"
    set color green
  ]
]]
end

to reproduce
if age > 10 [
  if energy > 30 [
    if random 100 < 70 [
      hatch 1 [
        set energy 5
        set age 0
        rt random 360
        fd 1
      ]
      set energy energy - 10
    ]
  ]
end

What happens to carry-capacity now?

Additional things to try:

1. Add sliders to let users control some of the variables which are now constant in the code.
2. Add a second level consumer that eats the red fish.
3. Add two second level consumers that compete for the red fish.

The model “Ocean Food Web” is a version of the model with some of these options. Notice how plants are handled in the model.